User's Guide™

TO CP/M® SYSTEMS & SOFTWARE

NO. 4 $4

CP/M Plus **SuperCalc** **Disk Utility** **dBASE II**

Multi-Reasons to Choose CompuPro's Multi-User System 816/C™

1. **Run *any mix* of 8 or 16 bit software at any terminal.**

2. **Lots of storage.** 384K of RAM (expands to 1 MByte), 2.4 million (expands to 4.8 million) characters of floppy disk storage, provisions for hard disk storage.

3. **Nine serial ports** for terminals, printers, modems, etc.

4. **Unparalleled single-user operation,** as well as optional expansion from 3 users to 16 users.

5. **Superior warranty.** 1 year limited warranty.

6. **Bundled software.** System 816/C runs over 3,000 CP/M® programs, and comes with MP/M™-816™, CP/M 2.2, CP/M-86, M-Drive™, SuperCalc™ (Sorcim's powerful 16 bit spreadsheet), and dBase II™ data manager.

7. **Ultra-high performance.** As *Interface Age* said, "The (8 bit) 8085 was more than a third faster than any 8 bit micro we have tested to date...the (16 bit) 8088 (was) almost twice as quick as the identically-engined IBM Personal Computer." You will appreciate this extraordinary level of performance when deadlines loom and time is of the essence.

System 816, MP/M 816, and M-Drive are trademarks of CompuPro; CP/M, CP/M-86, and MP/M are trademarks of Digital Research; SuperCalc is a trademark of Sorcim; dBase II is a trademark of Ashton-Tate. **Terminals courtesy Qume Corp.**

CompuPro®

CompuPro division, Godbout Electronics,
Box 2355, Oakland Airport, CA 94614

Authorized CompuPro Systems Centers specialize in business, industrial, and scientific microcomputing. They can show you why System 816/C represents the highest expression of 8 and/or 16 bit multi-user computing.

Circle reader service number 3

User's Guide
TO CP/M® SYSTEMS & SOFTWARE

User's Guide **Volume 1, Number 4 June 1983.**
User's Guide (ISSN applied for) is published six times a year by TUG, Inc., Box 3050, Stanford, CA 94305. Application to mail at second-class postage rates is pending at Woodside-Redwood City CA.

Entire contents Copyright © 1983 by TUG, Inc. All rights reserved. Reproduction of material appearing in *User's Guide* is forbidden without explicit permission. Send all requests to The Editors, Box 3050, Stanford, CA 94305.

SUBSCRIPTION INFORMATION:
Single copy price $4.00 ($5.50 sent first-class). Subscription for one year is: U.S. — $18 if payment accompanies order, $21 if subscription is billed. Add $10 for first-class delivery.
FOREIGN - $24 U.S. for one year (surface mail) if payment accompanies order, $29 U.S. if subscription is billed. Inquire about rates for airmail delivery outside the U.S.
Inquiries about bulk, retail sale, and group rates are encouraged. Address subscription inquiries to *User's Guide*, Box 3050, Stanford, CA 94305. Allow 6-8 weeks for processing of order.
POSTMASTER: Please send address changes (Form 3579) to User's Guide, Box 3050, Stanford CA 94305.

About the Cover

This issue's cover art is Brueghel's "Census at Bethlehem" with modern touches by Sahnta Pannutti. The editorial summarizes a "census" of the CP/M marketplace. You can measure the computer industry in terms of market share, but can you measure the health of the industry? To paraphrase Emerson, the true test of a civilization "is not the census, nor the size of cities, nor the crops, but the kind of man the country turns out." Will the current trends in the computer industry "turn out" more productive users? At least the census-takers can now use highly-productive tools like SuperCalc and dBASE II.

CP/M and CP/M Plus are trademarks of Digital Research.
SuperCalc is a trademark of Sorcim.
dBASE II is a trademark of Ashton-Tate.
User's Guide and Impatient User's Guide are trademarks of TUG, Inc.
User's Guide is not affiliated with Digital Research or any hardware or software manufacturer or distributor.

Contents

Vol.1, No.4 June 1983

Editorial... **Appropriate Technology** 2

For Starters... **Exploring dBASE II** 5
Data base management from the ground up.

Tutorial... **A Guide to SuperCalc® : Part Two** 8
by Tony Bove & Cheryl Rhodes
Check register, sales forecast, mortgage schedule and other spreadsheet applications.

An Impatient User's Guide... **SuperCalc SuperSummary** 23
by Tony Bove & Cheryl Rhodes
A quick reference guide to SuperCalc commands and functions.

Overview... **CP/M Plus® (CP/M® 3.0)** 29
by Steve Rosenthal
CP/M Plus for eight-bit systems looks a little like MP/M (multi-user version) with all its features, yet it is still compatible with earlier versions of CP/M.

Reference & Summary... **CP/M Plus® Operating Tools** 37
by Kelly Smith with Bove & Rhodes
Commands and utilities for banked and non-banked CP/M Plus (CP/M 3.0) systems.

Public Domain... **A Reference Guide to DU (Disk Utility)** 53
by Kelly Smith
An impatient user's guide to DU, a famous disk fixer from the free public domain libraries.

Keep Ahead of the Jargon Dept.... **Floppy Disks** 64
by Steve Rosenthal
What they're made of, and how to care for them. Floppy disks are unveiled and explained.

System Overview... **The CompuPro S-100 Computer** 74
by Allan Lundell
Near-excellence in hardware from Bill Godbout's Oakland Airport hangar.

Letters... 20

Product Announcements... Systems, software, peripherals 28

Advertiser Index... 47

EDITORIAL Tony Bove & Cheryl Rhodes (editors),
Kelly Smith, Steve Rosenthal (contributing editors)
CIRCULATION William Bruneau
ART Sahnta Pannutti (cover)
Contributors: Phi Bach Pham, Lon Shoemaker
PRODUCTION Flora Peduncle, Phi Bach Pham Design,
Anita Rhodes, Carol Hartung, Lon Shoemaker
MARKETING Kent Wilcox
OPERATIONS Carol Hartung, Anita Rhodes
Special Thanks: Dave Parkinson (Technoweb),
Betsy Jeavons, Eugene Wulsin, Jane Smith

Typesetting by Bove & Rhodes with George Graphics, San Francisco, and Abra Type, Palo Alto. Camera work by Camera/Graphic, Palo Alto. Printed by TechnoWeb, San Jose.

Appropriate Technology

CP/M or MSDOS? They both run the same application software!
UNIX? Not much business software available yet.

Where is the computer industry heading? What's in store for the CP/M user? Are the latest sixteen-bit machines the wave of the future? Is eight-bit dead? Will other operating systems fare as well, or outlast, CP/M? Will MSDOS, PCDOS, or UNIX become the most popular operating system for micros?

As a computer user you may not be too concerned with the future, since you already have an excellent productivity tool. However, new computer buyers *are* concerned with the future, waiting to buy in at the "right moment." While buyers should look at the problems of existing systems as well as promises of future products, computer users should be concerned with future expansion of their systems to make them compatible with future offerings.

Get several operating system experts together on a panel, and they'll disagree on which is the best system. However, good design or appropriate technology do not always produce the leading product. More often than not, marketing efforts produce the leading product, and software marketing efforts today in the micro industry are in capitalizing on what is *already out there* — the installed base.

As one marketer put it, "the installed base is IBM, the PCDOS system is the most popular because it comes free with the system and it is everywhere." Another disagreed: "CP/M is the obvious leader because there are over 3000 useful business application programs available for CP/M, and CP/M runs without compatibility problems on a variety of machines."

Meanwhile, UNIX is panned as a business tool. "There are no business programs for UNIX."

What are these systems, in nontechnical terms? PCDOS is the system that comes with the IBM PC. MSDOS is the PCDOS look-alike produced by Microsoft, the company that wrote PCDOS for IBM.

CP/M is the system we know most about, because it has been in use for over eight years. Vanilla CP/M (CP/M-80) runs on eight-bit systems; CP/M-86 runs on sixteen-bit systems like the IBM PC (and is therefore a competitor to PCDOS and MSDOS), and CP/M-68 runs on the 68000 chip, which makes it a competitor to UNIX.

UNIX has been around for the longest time, and is a product of the technology of the late sixties and early seventies. UNIX was invented by Bell Labs for use with DEC PDP-series minicomputers (minicomputers are the larger machines that support many users and cost $50,000 or more). UNIX is a *timesharing* system, which means it controls one computer for many users. Since UNIX is becoming available on powerful microcomputers that sell for above $5,000, it has the potential to become a popular operating system.

8-bit Systems, CP/M Plus

CP/M sits well in the eight-bit computer marketplace. It is, by everyone's measurement, the most widely used operating system, with over 1,000,000 installed systems. CP/M is available to run on computers from at least 700 different computer manufacturers, and at last count there were over 3000 applications programs for CP/M systems — mostly for small business and professional applications.

What this means to a professional or small business computer user is *compatibility*, *usefulness* and *choice*. Compatibility means you can run any of the 3000 programs on your computer, and you can share your data with users of over 700 different types of computers. You can even run the same programs on all 700 different computers.

Usefulness means the software is already available and supported by major software vendors. You do not have to wait a year or more for excellent software.

Choice means you can choose among 700 different types of computers, ranging in size and in price from as little as $500 to as much as $25,000. You can mix and match components such as printers, modems and terminals. You can use one type of computer at home, another at the office, and use the data with even another type of computer, as long as they all run CP/M.

The eight-bit world is thriving with low-cost portables and systems based on the Z80 chip. Radio Shack's Model 4 is a small business system for under $2000 that runs the newest version of CP/M-80 called CP/M Plus. Osborne just announced the $2500 Executive portable that has expanded memory (128K) to run the banked version of CP/M Plus. Rumor has it that IBM may offer a similar machine (128K internal memory with CP/M Plus).

Xerox has just announced the 16/8 system offering CP/M Plus, CP/M-86 and MSDOS as operating systems. This system is right in line with the trend to provide both sixteen-bit and eight-bit computing. Digital Equipment Corp. (DEC) may be the largest contender to IBM with its Rainbow computer that also provides eight-bit and sixteen-bit computing, running a powerful DEC version of CP/M. Dynabyte's Monarch is another well-designed system with several processors available to run eight-bit or sixteen-bit applications.

16-bit Systems, IBM

A product endorsed by the industry giant IBM has an advantage in the computer marketplace over products that are ignored by IBM, whether or not the product has a better design. MSDOS, the operating system designed by Microsoft and adapted for the IBM Personal Computer as PCDOS, is the current leader because it is distributed free with the computer. IBM also recognized the importance of CP/M-86 and UCSD Pascal, but these products were priced beyond the range of consumers (IBM's version of CP/M- 86 was $250 per computer). Digital Research, producers of CP/M, responded recently by providing an improved version of CP/M-86 for $60.

MSDOS, although not completely compatible with PCDOS, is close enough to allow another manufacturer's computer to act like an IBM PC. However, a lack of standardization in screen controls and graphics makes MSDOS less compatible on different machines. MSDOS does not exist in the eight-bit world, and MSDOS data files cannot be used with computers running eight-bit software unless they are converted to a CP/M format.

CP/M-86, on the other hand, provides the closest compatibility for a manufacturer's computer to both IBM PCs running CP/M-86 and other computers, such as the DEC Rainbow, the Radio Shack Model 16, the new Texas Instruments and Wang desktop computers, the new Xerox model 16/8, the new Casio and Fujitsu computers, and many other systems. The GSX graphics subsystem of CP/M-86 is a proposed software standard for all graphics on micros, since it conforms to European standards for all systems, existing minicomputer graphics systems, and videotex standards endorsed by AT&T.

CP/M-86 also offers data compatibility with all CP/M-based eight-bit systems — Xerox, Zenith, North Star, CompuPro, Vector Graphic, Wang, DEC, Osborne, Radio Shack (CP/M option), Apple (running CP/M card), and the other 700 manufacturers offering inexpensive CP/M machines.

CP/M-86 users can grow to use Concurrent CP/M-86, and MSDOS users can grow to use XENIX, Microsoft's version of UNIX. Both growth paths represent appropriate future trends in systems and software, and it remains to be seen which path is the more popular one.

No doubt both will be in strong positions for a long time. Software producers are ignoring the differences between the systems and are writing software for both. The most popular CP/M programs (WordStar, dBASE II, Accounting Plus, SuperCalc, MultiPlan, etc.) all have MSDOS/PCDOS counterparts. This is due to design similarities between MSDOS and early versions of CP/M for eight-bit systems. MSDOS was described as a "CP/M-lookalike" when it first appeared, and many CP/M programs were changed to run under MSDOS. However, data and text files were not transferable without a conversion process.

Programs could be written for both systems if they were written in Microsoft's BASIC or in other languages such as Pascal, Fortran, Cobol, C, PL/I, and Forth. However, many excellent business programs are written in CBASIC, and most of these have now been translated to CP/M-86 thanks to Digital Research's new CBASIC-86. Eight-bit CBASIC programs can be translated to sixteen-bit CP/M-86 systems with practically no changes.

For many users, the question of using MSDOS or CP/M-86 doesn't matter because the same popular programs are available for both systems. For others, the differences between the two systems require a decision to use one or the other.

At this time CP/M-86 is the clear underdog for IBM Personal Computers, since PCDOS has already captured the largest share of installed systems. However, the freely-distributed PCDOS may only be a starting point for serious users who want to expand their systems in the future — perhaps to Digital Research's Concurrent CP/M-86, which lets you do more than one thing at a time with your IBM PC and many other sixteen-bit computers.

Future Compatibility

The technology is now available in Concurrent CP/M-86 to have a single user performing ten or twenty tasks simultaneously with a single computer. The next step in this evolution is to link these desktop "workstations" into a *local network* scheme that lets several computers share one large disk drive, one printer, etc.

The computer you buy today may not play in tomorrow's local network. In an automated office, every user has a personal computer that can share its data with other computers, communicate among computers using "electronic mail," and share the more expensive printers and hard disks.

Some of these activities require *concurrent* systems running on personal computers. A concurrent system is one you use by yourself to perform multiple tasks. Rather than waiting for printouts to finish or programs to perform routine tasks, you can start these "jobs" on your personal computer and let them run concurrently, while you do something useful with the computer.

The design also calls for standard file structure. In corporations, dozens of different computers have to share programs and data. CP/M provides a file structure consistent with the eight-bit world, with the ability to handle password-protected files.

With concurrency, CP/M-86 provides a mock timesharing environment for *one* user, making that one user more productive. UNIX, when transported to a single-user machine, also offers concurrency and the powerful tools associated with the UNIX "environment."

However, powerful tools do not always sell. UNIX is hard to learn, largely because its prophets are wizard programmers who don't have time to explain in simple terms how to use the system. UNIX is a wonderful environment for writing programs, but there is a serious lack of business software for UNIX.

UNIX does offer one important future trend, which it shares with MSDOS and PCDOS, but not with CP/M: a tree-structured directory of files, where individual files can be stored within topic-oriented directories. This type of file directory structure helps you organize lots of files stored on a hard disk.

CP/M doesn't have this organization because CP/M was designed for floppy disk environments. The structure is "flat" because file space is limited on floppies. However, CP/M is designed to be improved by system packagers and by sophisticated users themselves. You can buy off-the-shelf file management programs, and library/archive utilities are available for free in the CP/M software public domain.

The CP/M and UNIX worlds will come closer together, with the same application software offered for both. This trend is inevitable because many programmers are switching to a more universal programming language for systems programming: C.

Programs written in C can be translated to almost any system. C, a language from the UNIX world, is now the most popular systems language, used extensively by Digital Research and many other companies. The latest versions of CP/M itself are written in C, and a C compiler is included with CP/M-68 to compile programs for both CP/M and UNIX systems. These tools will provide compatibility between widely different operating environments.

Integrated Applications

The latest design trend in application software is the *integrated package* that provides several types of application programs in one package with a universal set of commands.

Several characteristics are prominent in these designs: the use of multiple "windows" on the screen, with each "window" showing a different but possibly related application. For example, a spreadsheet could be in one window, a graph of the spreadsheet's results in another, and some text explaining the spreadsheet in yet another window.

VisiOn, a product announced (but not yet available) from VisiCorp, lets you put many Visi-series programs into one controlled environment that shows windows of applications. VisiOn, from VisiCorp, also lets you use the newest "mouse" device that moves the cursor on the screen when you roll the "mouse." (VisiOn will not be available until the end of 1983.)

Apple's new Lisa computer has multiple windows and the "mouse" device, but the software does not perform as well as expected at this time. For example, the graph in one window would not change immediately after you change a number in the spreadsheet window. (There is a backlog of Lisa computer orders which make the computers unavailable until early 1984.)

Lisa can provide different applications under one roof because the programs share a common data file structure. The data can be transferred easily from program to program. This idea is not new — CP/M defined a standard file structure in use now for eight years, and "data dictionaries" (more complex structures for data base files) have been around the minicomputer generation.

A concurrent version of VisiOn will let you display a spreadsheet and a graph of the spreadsheet, and automatically change the graph while changing values on the spreadsheet. Meanwhile, the concurrent operating system could be receiving electronic mail and sorting a data base.

Digital Research designed Concurrent CP/M-86 to have a "real-time nucleus" that allows messages to pass from one running program to another program running simultaneously. As a result, Concurrent CP/M-86 can provide added capabilities to Lisa-like systems like VisiOn. VisiCorp has an agreement with Digital Research to help produce a version for all CP/M-86 machines.

Integrated software systems like Starburst from MicroPro (which puts the WordStar family of products under one "menu"), VisiOn from VisiCorp, DesQ from Quarterdeck (which lets you put any set of programs under one "menu"), T/MAKER III and Context MBA give users one environment in which most operations occur.

The same environment means that the same text editing commands can edit spreadsheets and be used to update the data base. Numbers in rows and columns within other documents can be graphed. You need to type the information once, and then it is available to the other programs without retyping.

A common file structure for all programs is necessary for this to occur, and concurrent systems offer the only way to perform these operations simultaneously. These types of systems are the wave of the future.

You can measure the computer industry in terms of market share, and compare Apples to IBM Personal Computers in terms of business applications. Can you measure the health of the industry? To paraphrase Emerson, the true test of a civilization "is not the census, nor the size of cities, nor the crops, but the kind of man the country turns out." Will the current trends in the computer industry "turn out" more productive users?

We like systems that adapt to humans rather than requiring humans to adapt to them. Nearly every computer system today provides the "open space" between the user and the required system commands — space to be filled by sound integrated application "super-programs" that remember your operational habits (your "user profile") and run your programs simultaneously while providing easy access to the information. It is these super-programs that we look forward to describing in these pages.

Exploring dBASE II

A Beginner's Approach to Data Base Files, and a Perspective on Data Base Management, Using dBASE II® as a Launching Pad

by Tony Bove & Cheryl Rhodes

dBASE II is a trademark of Ashton-Tate.

Beginners are easily intimidated by data base management systems. The software can do so much, yet the new user can do so little without extensive training.

Training consists of learning how to use the system's command language, defining your application, and deciding in advance what types of data will be stored and how data should look upon retrieval. Knowing these details is looked upon as a necessity to learning data base management. Very early in this training, many new users fall victim to the "requirements definition."

The "requirements definition" forces you to define the way you use information now, and how you will use it in the future. What information will you need next year? What form should it take? How much room should you set aside for future information?

These are tough questions to answer. You may decide now to structure your information a certain way, and find that later you will have to reorganize the data base. Such reorganizations can be easy with some data base management systems, and hard with others. It might happen that the only way to reorganize your data may be to actually re-type all of it.

In the dark ages, corporate data processing often meant centralized management of computing. Stringent rules on information access allowed political "pecking order" struggles to strangle productivity. Managers in the MIS departments were looked upon as the high priests of jargon, with their nearly religious requirements and inscrutable reports. Data entry clerks typed inscrutable data; other clerks prepared entries on paper, writing part numbers and descriptions into little boxes on data entry cards, in reverse so that the descriptions were right-justified for the data entry clerks...

The systems were highly mysterious. Designers programmed in a special control language and invented extraordinary systems, many of which actually worked; however, they spoke a mumbo-jumbo no one understood, and they often left behind unreadable documentation.

Avant-garde computerists in the seventies proclaimed that future personal computers would have the power to present data base information directly to executives and professionals who have no computer experience or training.

These personal computers have arrived and are in use on desktops around the world. Some systems can be useful for people who could care less about the arcane mysteries of computing. Programs like dBASE II, DataFlex, Access Manager, WHATSIT, and other small computer data base systems can actually be used by anyone without programming experience, once they have some experience with using them.

This article is a quick primer for data basing with dBASE II, the popular data base management program from Ashton-Tate. dBASE II runs on all CP/M systems (including the IBM PC, XT and PC compatibles) and was distributed bundled with Osborne portable computers last Christmas.

What's In a Data Base File?

A filing cabinet is a paper "data base." Usually you keep papers inside file folders with labels. You find information about client JOHN SMITH by searching the labels for SMITH, JOHN? Maybe you don't... maybe you look John up by his account number, his social security number, or the last time you billed him.

For any of these methods and for many other methods, you use a *key* to look up JOHN SMITH, and what you unlock is a *record* containing the information about JOHN SMITH. The key may be his name (SMITH), his "customer ID number" or some such thing (e.g., "SMI965-Q"), his social security number (174-56-8738), or the invoice number of the bill you sent him (#235).

dBASE II, like most "relational" data base systems, lets you look up records in a variety of ways. The easiest, at first, is by *record number*. As you type information into each record, dBASE II numbers the records starting at record #1. It is easy to find a record by its number if you know its number; however, as your data base grows, you soon forget which record has which record number.

How do you create a data base? Start with *why* you need one. Are you starting a mail-order business? Do you need to process orders? Are you a doctor, lawyer or other professional with clients or patients to be billed and kept track of?

Do you need to keep an inventory of materials while you build a house? As a loan officer in a bank or an insurance agent in a large insurance company, you may want to keep your own information about customers, rather than rely on the corporate data processing department. Or you may want to show on a computer screen all bank loan options or insurance plans, and perform "what if" calculations using the data.

These are tough questions. How can you visualize what you want when you have no experience with data bases? Think about what you use now, and the ways you "access" or find the information.

Do you use a cardbox file or rolodex to hold a mailing list? Do you use a filing cabinet to hold client or patient records? Do you periodically walk around the yard and count pieces of building materials, and record your count in a clipboard or notebook? Do you rely on the corporate DP department for monthly listings and weekly updates? Do you pull plans out of your briefcase and use a calculator to determine monthly payments?

Do you want to change your way of doing things? This is perhaps the hardest question to answer. Most people would answer yes, if only to keep in step with everyone who knows about computers. The answer should be yes only if the computer makes this transition painless.

Unfortunately for most users, the transition is never painless unless the user wants to change paper-bound habits. But how do you know unless you try?

Start with a simple data base: a list of clients. Pretend you are a professional contractor, and you need a list of clients, the companies they work for, and the amount each owes you for your contracting services. You ask the fundamental questions:

1. What information do you want to be able to look up?

In this case, you want the full name, address and phone number, and the company name and the contact's title in the company. In addition, you may want an alternate phone number and a comment about the person or company. Finally, you want the amount the company or client owes you.

2. What do you already know about the information, to use as a key or keys to looking things up?

You may want to look up records by last name or by company name. For example, you should find WILCOX if you know his name, or find WILCOX if you know his company's name (you should also find all names associated with his company). You might also want to find all ACCOUNT MANAGERS in all companies, or find ACCOUNT MANAGERS for his company only.

If you're looking up by last name, you should be able to find Ken Smith by looking up SMITH, and reading each SMITH until finding Ken Smith (after David Smith and Kelly Smith).

3. What do you want as output?

This question is not as important now, because most data base systems will produce printed reports according to your specifications. You need not specify the reports now.

However, you should know a little about what you want. Do you want mailing labels? Form letters? Sales summaries? Client or patient summaries? Detailed printing of each record? All of these?

You should also think about printing records in certain order. For example, you might want to produce mailing labels in zip code order. You may also need printed summaries listing last names alphabetically with accompanying phone numbers, or printed lists by company name in alphabetical order. Nearly every data base application needs a master printout that shows all information, sorted alphabetically by last name or some other key.

Phew! So many things to think about! Ahh, but focus on what you need for now: you need a list of contacts you can access by name or company, print by last name alphabetically, and print labels by zip code.

That's all you need now. You can expand the data base later to include more information. You can even re-structure an existing data base without destroying the data, and you can transfer the data to another more sophisticated data base, or even transfer it to a system other than dBASE II. In most cases you will never have to retype a data record. So why worry about the data base structure now?

Simply define it as you need it. The following dBASE II dialogue defines the client data base in a straightforward manner. All you have to do is type the following command to start dBASE II:

```
A)DBASE↵
```

The program asks for the date, which you can ignore by pressing RETURN if you wish (or you can supply a date if you wish), then it displays a period. You can now type dBASE II commands. We show commands in lower case to distinguish them from dBASE II messages, which we show as UPPER case letters:

```
.create clients↵
ENTER RECORD STRUCTURE AS FOLLOWS:
```

Before you enter a definition of how each record will look, you need to decide how many characters or digits to use for each item in the record. How many letters will be used for the longest last name? We use twenty — if the name is longer, we abbreviate the name. How many digits do you need to use in the amount owed, and how many of them are decimal places? We chose ten digits, two of them decimal places.

You must also choose names for each item. The item's name, known as a *field name*, starts with a letter and can have up to ten characters including letters, digits, and colons.

Here is our *structure*:

```
FIELD     NAME,TYPE,WIDTH,DECIMAL PLACES
001       company,c,25↵
002       lastname,c,20↵
003       first:mid,c,25↵
004       title,c,15↵
005       address1,c,30↵
006       address2,c,30↵
007       city,c,15↵
008       state,c,2↵
009       zip,c,6↵
010       phonenum,c,20↵
011       altnumber,c,20↵
012       amt:owed,n,10,2↵
013       comment,c,40↵
014       ↵

   INPUT DATA NOW? Y
```

At this point, if your dBASE II program is installed for your screen, you should see an input screen that looks like this:

```
   RECORD 00001
COMPANY       :
LASTNAME      :
FIRST:MID     :
TITLE         :
ADDRESS1      :
ADDRESS2      :
CITY          :
STATE         :
ZIP           :
PHONENUM      :
ALTNUMBER     :
AMT:OWED      :
COMMENT       :
```

If your version of dBASE II is not installed for full screen features, you may only see the first item (COMPANY), and you can't use many of the handy editing features of dBASE II; however, the program is still very useful and can be run on almost any terminal without full screen features. It is much better to install the full screen features.

Following each field name is a "picture" of each field, showing the number of characters you can type before you fill it. You can type less than the full number, but you can't type more than the allowed number of characters. Press RETURN to end fields with less than the full number of characters. When you finish record 1, press RETURN at the top of record 2 to stop entering data:

```
   RECORD 00001
COMPANY       : User's Guide to CP/M↵
LASTNAME      : Bove↵
FIRST:MID     : Tony↵
TITLE         : Editor & Publis
ADDRESS1      : P.O. Box 3050↵
ADDRESS2      : ↵
CITY          : Stanford↵
STATE         : CA
ZIP           : 94062↵
PHONENUM      : 415-493-1232↵
ALTNUMBER     : ↵
AMT:OWED      : 240.00↵
COMMENT       : Article on dBASE II.↵

   RECORD 00002
COMPANY       : ↵
```

Did your keyboard "beep" at you? If your keyboard is equipped with a "bell" or "beep," dBASE II uses it to tell you that you typed the full amount of characters in a field (TITLE in the example above, and STATE in every case). If you don't like this "beep" you can turn it off with this command:

```
.set bell off↵
```

The structure of your data base file is simple and straightforward. You can see it by typing this command:

```
   .display structure↵
   NO DATABASE FILE IN USE, ENTER FILENAME: clients↵

   STRUCTURE FOR FILE: CLIENTS.DBF
   NUMBER OF RECORDS: 00001
   DATE OF LAST UPDATE: 00/00/00
   PRIMARY USE DATABASE
FLD        NAME         TYPE  WIDTH  DEC
001        COMPANY      C     025
002        LASTNAME     C     020
003        FIRST:MID    C     025
004        TITLE        C     015
005        ADDRESS1     C     030
006        ADDRESS2     C     030
007        CITY         C     015
008        STATE        C     002
009        ZIP          C     006
010        PHONENUM     C     020
011        ALTNUMBER    C     020
012        AMT:OWED     N     010    002
013        COMMENT      C     040
** TOTAL **             00259
```

This structure is actually a simple *data dictionary*. You have actually defined a simple but typical data base. dBASE II lets you define this in less than five minutes.

You may be able to see how your business or profession could use such a tool. dBASE II has its own *command language* you can use to write small programs to control data entry or other activities.

Next issue will focus on dBASE II and ways to define and organize your information so that you can find it quickly and prepare the appropriate reports, mailing labels, or form letters.

A Guide to SuperCalc: Part Two

Check Register, Sales Forecast, Mortgage Schedule, and Other Spreadsheet Applications

by Tony Bove & Cheryl Rhodes

SuperCalc, from SORCIM Corp., is a popular program for calculating rows and columns of numbers. It is similar to VisiCalc, that "newfangled program that makes an Apple II computer worthwhile in business." SuperCalc was created for use on CP/M systems. It was the first such program that could run on different computers using different display screen characteristics, and it spawned a second generation of "calc" programs.

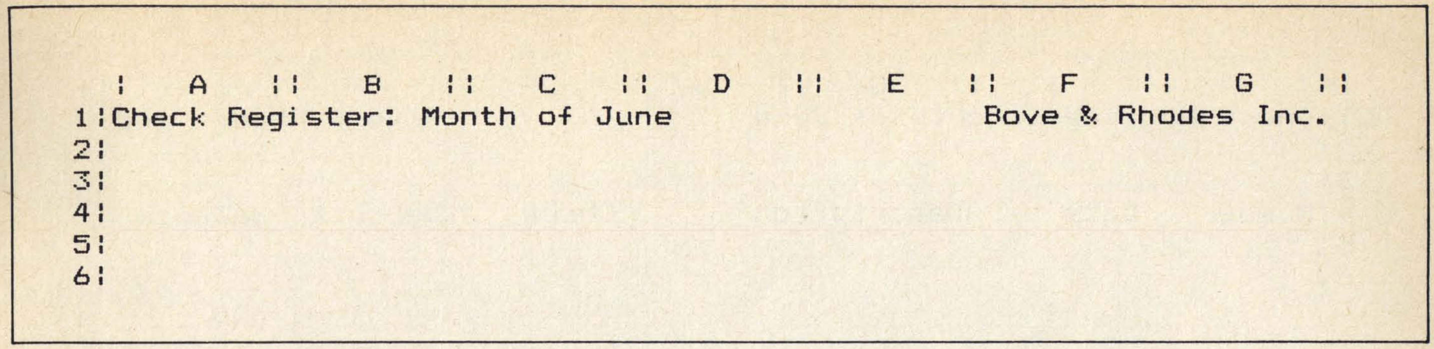
"Calc"-type row and column calculations may rank second, behind word processing and slightly ahead of data base management, in the list of most popular computer applications. Some computer manufacturers recognize that these *spreadsheet* programs, used regularly by a majority of computer users, should be provided with their computers. SuperCalc is offered "bundled" with Osborne portable computers, Xerox 820, and others.

With SuperCalc comes SuperData Interchange®, a program that lets you transfer SuperCalc spreadsheet data to other programs and systems. SuperData offers a format (the SDI format) for storing data which lets you move spreadsheets to other systems and to other programs. SuperData also supports the DIF format (used by VisiCalc® from VisiCorp, and other programs), and the CSV (comma separated values) format used with DataStar® (from MicroPro Intl.), dBASE II® (Ashton-Tate), and many BASIC programs.

NOTE

In this article we use the ↑ (up-arrow) symbol to stand for the Control key on your keyboard (sometimes labeled CTRL, CNTL or ALT), and we use the ↩ symbol to stand for the RETURN key (sometimes labeled CR, RET or ENTER). The RETURN key is usually a larger key located on the right side of your keyboard; the Control key is usually a smaller key on the left side.

To use SuperCalc, you type shorthand commands at your keyboard. SuperCalc displays the full name of each com-



```
  ¦   A   ¦¦   B   ¦¦   C   ¦¦   D   ¦¦   E   ¦¦   F   ¦¦   G   ¦¦
 1¦Check Register: Month of June                Bove & Rhodes Inc.
 2¦
 3¦
 4¦
 5¦
 6¦
```

Figure 1. *Opening line of the spreadsheet, used as a comment in SuperCalc file directory displays.*

mand and option as you type the shorthand version, and displays many status messages on your screen. To differentiate between what you type and what SuperCalc displays, we use *THIS TYPEFACE* to show the letters you should type, and THIS TYPEFACE to show full-length commands and messages displayed by SuperCalc or the system.

This article is part two of a guide to SuperCalc. In part one (*User's Guide* #2), readers learned how to start SuperCalc, edit a sample spreadsheet, and print the spreadsheet. These lessons show how to create your own spreadsheets — a budget, a comparative income statement, a cash flow statement, and an income tax worksheet. Each application highlights a feature of SuperCalc or an advanced technique.

Starting SuperCalc, and Moving the Cursor

Start SuperCalc by typing *SC*↵ followed by another RETURN to skip the help message. A blank worksheet should appear on your screen, with row numbers on the left side and column letters across the top.

The SuperCalc cursor is in cell A1 (column A, row 1). On some computers and terminals, you can move SuperCalc's cursor using the directional keys (the arrow keys). On those that do not have arrow keys (or the version of SuperCalc has not been installed for the specific terminal or computer you are using), the cursor movement keys are the same as the ones used in WordStar:

↑*X*	Move down one row (line).
↑*E*	Move up one row (line).
↑*D*	Move to the right one column position.
↑*S*	Move to the left one column position.

Creating and Using Spreadsheets: A Monthly Check Register

To get a feel for creating a spreadsheet, start with a simple check register that records checks and deposits for a one-month period.

Start with a label for the entire spreadsheet in cell A1. Get in the habit of using a label in cell A1 as the spreadsheet's title. SuperCalc will display this title, or whatever is in cell A1, as a comment for the spreadsheet when you ask SuperCalc to display a directory of spreadsheet files.

Type the label by first typing a double quotation mark (*"*), then typing the label. Press RETURN to end the label and record it on your worksheet. There is no need to use a quotation mark at the end of the label; if you do, the mark will become part of the label.

If you forget the initial quotation mark, SuperCalc assumes you are typing a numeric value, a formula, or a special function. SuperCalc will display FORMULA ERROR if you forget the quotation mark (unless you accidentally type a correct formula!). If you get this message after pressing RETURN, use your backspace key (or left-arrow key) to return to the beginning of the line, and type the initial quotation mark.

This sometimes happens to beginners and causes them to distrust the program at first.

Your label can be as long as you want. SuperCalc lets you type very long labels. Since you won't be using the rest of row 1 (cells B, C, etc. will not be used), SuperCalc lets the long label in cell A1 extend beyond its boundary, as shown in figure 1.

You can add more labels if you wish; we keep typing to a minimum in these "hands-on" examples. You may want to add your name or your company's name as a label starting at cell E1 or F1.

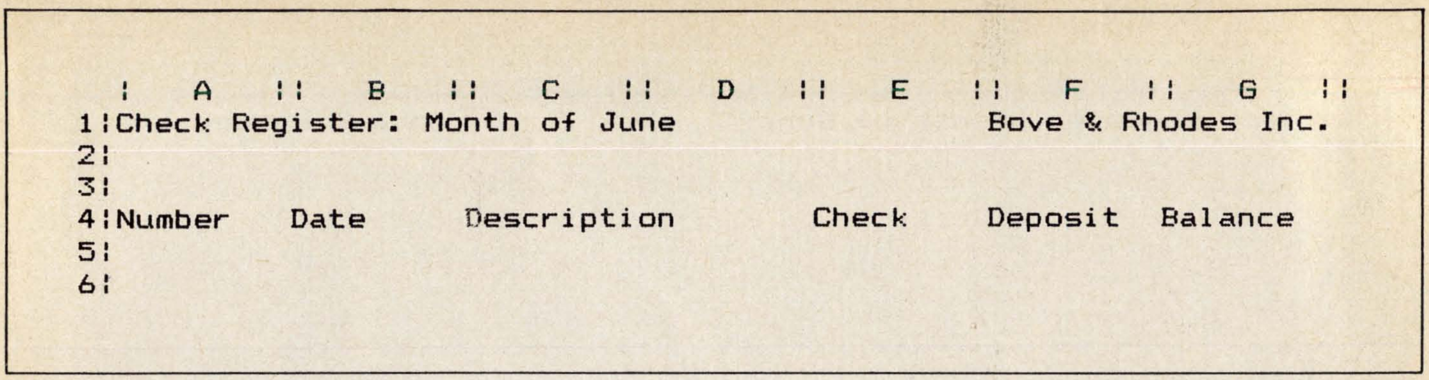
Move your cursor down to cell A4 and type another label: *"Number* (the word Number should appear on your screen). Press RETURN, and your cursor should move down to the next cell (A5). Move your cursor up one row, and over to cell B4.

Type the label *"Date* and press RETURN. The cursor should move horizontally to the next cell on the same row — cell C4. When you press RETURN, SuperCalc moves the cursor in the direction you last moved.

Type another label in cell C4: *"Description.* Press RETURN, and the cursor should move to cell D4. Press RETURN again to leave D4 blank. In cell E4, type the label *"Check*, and type *"Deposit* in F4 and *"Balance* in G4. Your screen should look like figure 2.

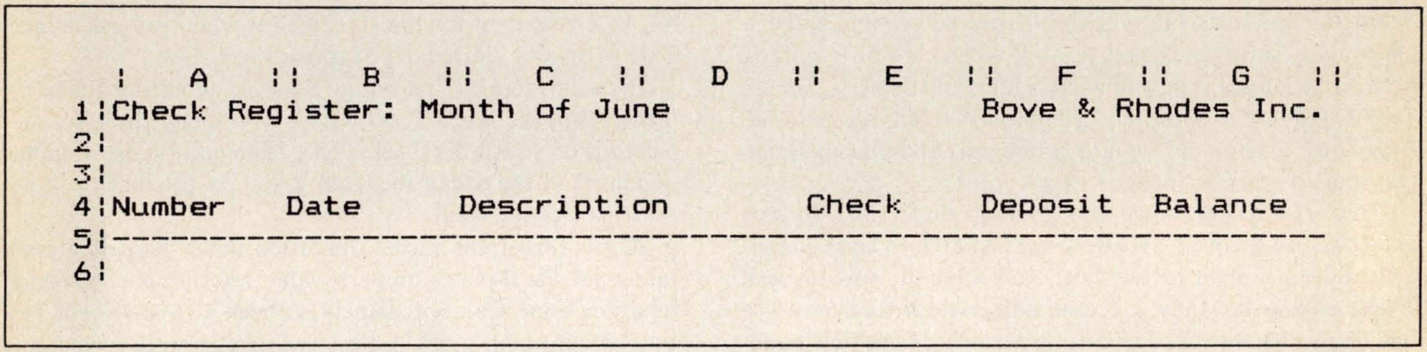
Replicating an Entry

Move the cursor to the cell below Number (cell A5). Type a



```
  |    A    ||    B    ||    C    ||    D    ||    E    ||    F    ||    G    ||
 1|Check Register: Month of June                Bove & Rhodes Inc.
 2|
 3|
 4|Number   Date     Description       Check    Deposit  Balance
 5|
 6|
```

Figure 2. *Check register with labeled columns.*



```
  |    A    ||    B    ||    C    ||    D    ||    E    ||    F    ||    G    ||
 1|Check Register: Month of June                Bove & Rhodes Inc.
 2|
 3|
 4|Number   Date     Description       Check    Deposit  Balance
 5|---------------------------------------------------------------
 6|
```

Figure 3. *Headings underlined using the Replicate command.*

quote and a line of dashes (e.g., "— — — — —). Type at least nine dashes, and more if you wish. Press RETURN to enter the dashes as a label. If the cursor does not stay in cell A5, move the cursor back to cell A5.

Now type a slash and an *R* for the Replicate command. SuperCalc displays the following:

```
From? (Enter Range)
12>/Replicate,
```

To see the possible answers, type a question mark for SuperCalc's AnswerKey® feature. SuperCalc explains what a "range" is:

```
A "range" is a row, column, cell or block.
A "row" is a number from 1 through 254.
A "column" is a letter (pair) from A through BK.
A "cell" is a column followed by a row, for example: 'J10'
A "block" is two cells, separated by a colon (:).
```

Press any key to return to the SuperCalc command line.

In a Replicate command, you can press RETURN for SuperCalc to use the current cell as the range. Press RETURN, and SuperCalc displays another message:

```
From? (Enter Range)
12>/Replicate,↵
To? (Enter Range), then Return; or "," for Options
15>/Replicate,A5,
```

In almost any SuperCalc command you can refer to an actual cell by pressing the Escape (ESC) key on your keyboard, and moving the cursor to the cell. Press your ESC key, and A5 should appear on the command line. Move your cursor to the right one cell, and B5 should appear on the command line.

You are replicating one cell, the dashed line, to several cells — to a *block* of cells. To specify a block, type a colon (:). SuperCalc displays the range B5:B5 on the command line. Move your cursor to the right several cells until the range is B5:G5. Press RETURN, and SuperCalc replicates the dashes across the range of cells, creating a line of dashes under the heading as shown in figure 3.

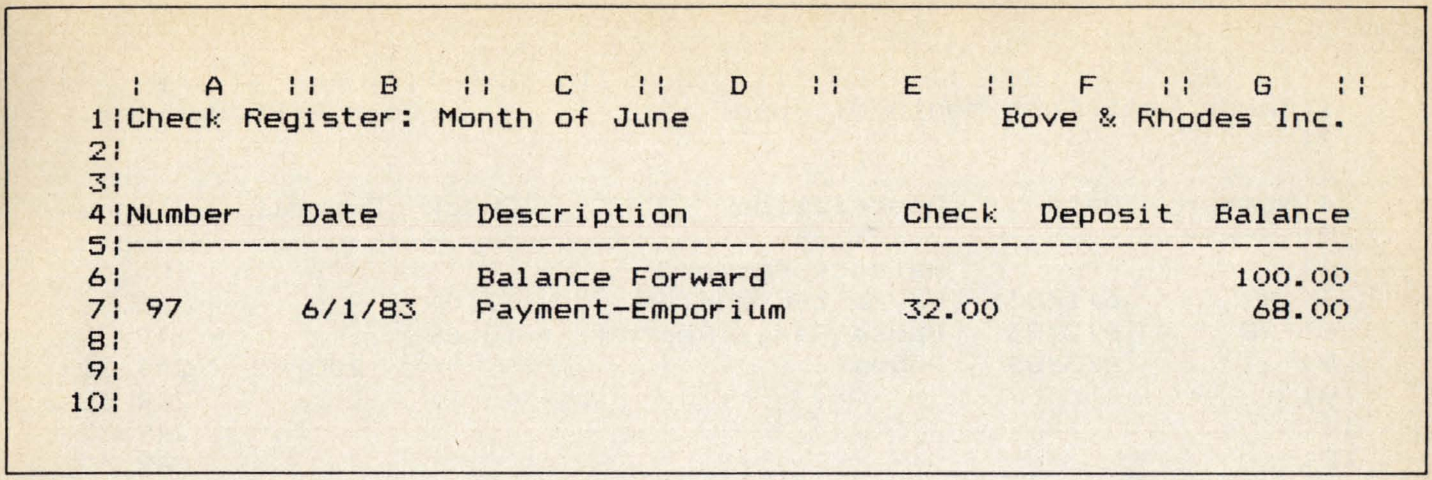
Type the label *"Balance Forward* in cell C6, and move the cursor to cell G6. Type a decimal number for the dollar amount of the beginning balance of your checking account. For example, if you are starting with $100, type *100* and press RETURN.

Entering Formulas

Each cell can contain a label, a number, or a unique formula to calculate its contents. Move your cursor to cell G7 and type the following formula (you can type the formula in lower case letters):

```
> G7
Width: 9      Memory: 29    Last Col/Row:G6       ? for
HELP
1>g7+f7-e7↵
```

The formula consists of the coordinates *g7*, *f7* and *e7* in a numeric expression that adds the values in cells G7 and F7 and subtracts the value in E7. The formula is: Beginning Bal-



```
  ¦   A   ¦¦   B   ¦¦   C   ¦¦   D   ¦¦   E   ¦¦   F   ¦¦   G   ¦¦
 1¦Check Register: Month of June                Bove & Rhodes Inc.
 2¦
 3¦
 4¦Number   Date     Description           Check  Deposit  Balance
 5¦---------------------------------------------------------------
 6¦                  Balance Forward                         100.00
 7¦ 97      6/1/83   Payment-Emporium      32.00              68.00
 8¦
 9¦
10¦
```

Figure 4. *Check register with formatting to improve its appearance: integer numbers in column A, decimals in columns E, F and G, and right-justified labels in cells E4, F4 and G4.*

ance + New Deposit - New Check.

Let's test the formula. Move your cursor to cell A7. Type the number *97* for the check number, *"6/1/83* for the date, *"Payment-Emporium* for the description, and *32* for the check amount. SuperCalc should automatically calculate cell G7 and display 68, the balance after writing the check. The formula should work.

NOTE

If you get the number .07 in your "Date" column, you must have divided 6 by 1 and then by 83. In other words, you typed the date without an initial quotation mark. Go back to the cell and re-type the date as *"6/1/83*

Formulas can consist of any of the following in acceptable combinations to form expressions:

- Numbers.
- Cell coordinates.
- Signs to signify addition, subtraction, multiplication, division, and exponentiation (usually Shift 6).
- Parentheses to indicate order of arithmetic operation.
- Special functions to calculate a square root, a logarithm, sine, cosine and tangent values, an average of a range of numbers, the net present value of a series of cash flows, etc.

Some examples of formulas:

C12(C13-D13)*
— The contents of cell C12 multiplied by the result of subtracting D13 from C13.

SUM(B2:B10)
— The sum of all values in column B from row 2 to row 10.

SQRT(SUM(B2:B10))+INT(A3)
— The square root of the sum of B2:B10, plus the rounded integer value of the contents of cell A3.

Formatting Cells, Rows and Columns

Some formatting can make your cells look better. Right now, your check number is probably too close to the date. This is because SuperCalc places numbers right-justified within a cell. The first change you can make is to format the entire A column.

Move your cursor to column A, any row, and type */F* for the Format command. SuperCalc displays in response:

```
Enter level: G(lobal), C(olumn), R(ow), or E(ntry)
9)/Format,
```

Type a *C* to format an entire column, and press RETURN for the current column (column A). Your command line should look like this, with SuperCalc's message line displaying many options:

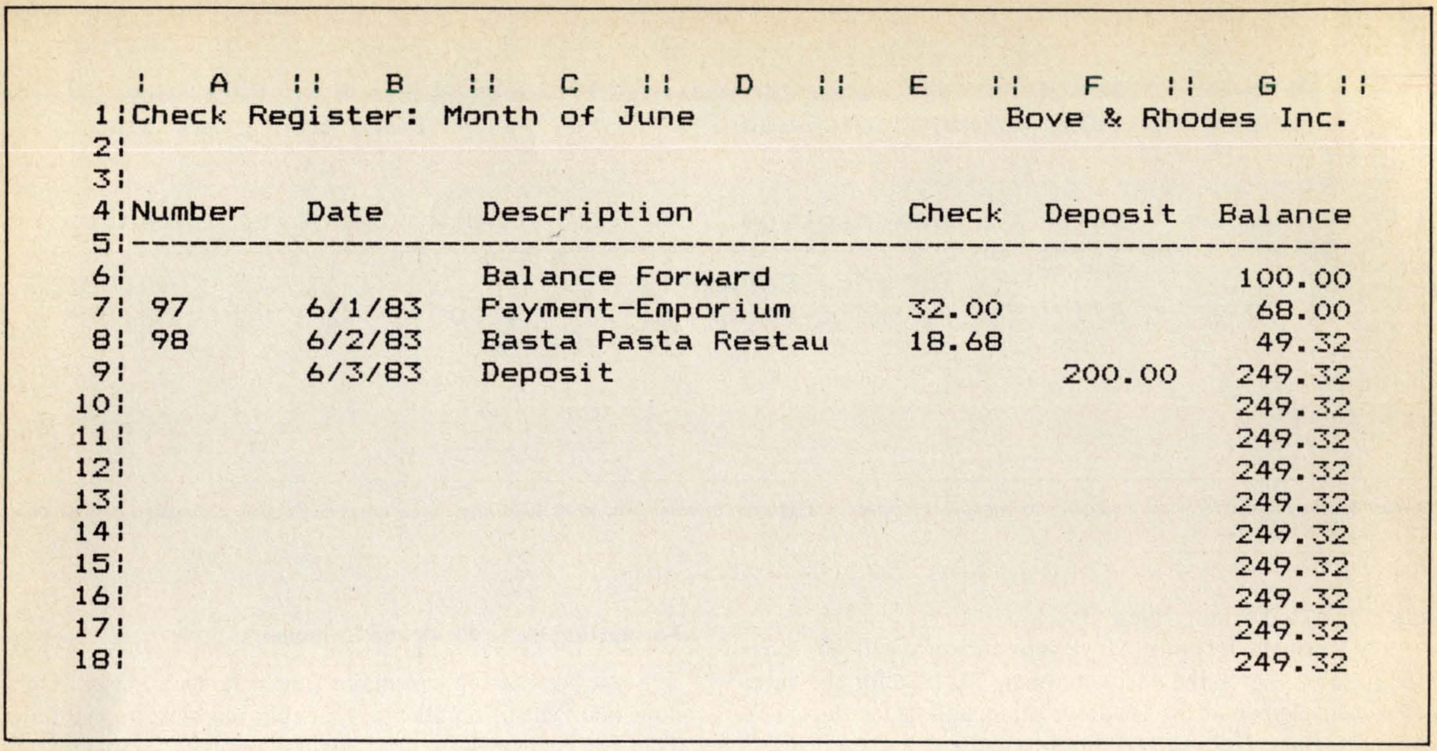
```
Define Formats: (I,G,E,$,R,L,TR,TL,*,D,column width)
18)/Format,Column,A,
```

If you press a question mark (the AnswerKey), SuperCalc displays a help message with a comment for each option. You will select the *L* option (for "L(eft)") to justify the numbers to the left margin of the column. To leave the help message, press any key. Type an *L* to select the "L(eft)" option, and press RETURN. The entire column is now left-justified for numbers.

A few more changes would improve the appearance of the spreadsheet. Move to cell E4 to change the formats of the heading cells E4, F4 and G4.

Type the format command */F*, then type *E* for a single entry (one cell). For the range, press RETURN, then type *TR* for the "T(ext)R(ight)" option. Press RETURN, and the cell's label should now be justified to the right margin. This formatting puts the heading directly above the number.

Use the same command in cells F4 and G4: */FE↵TR↵*. For each cell, SuperCalc displays the chosen formatting option above the message line (two lines above the command



```
   !   A   !!   B   !!   C   !!   D   !!   E   !!   F   !!   G   !!
  1!Check Register: Month of June                 Bove & Rhodes Inc.
  2!
  3!
  4!Number   Date     Description              Check  Deposit  Balance
  5!-------------------------------------------------------------------
  6!                  Balance Forward                           100.00
  7! 97      6/1/83   Payment-Emporium          32.00             68.00
  8! 98      6/2/83   Basta Pasta Restau        18.68             49.32
  9!         6/3/83   Deposit                          200.00   249.32
 10!                                                             249.32
 11!                                                             249.32
 12!                                                             249.32
 13!                                                             249.32
 14!                                                             249.32
 15!                                                             249.32
 16!                                                             249.32
 17!                                                             249.32
 18!                                                             249.32
```

Figure 5. *Typing checks and deposits into the automatic calculating check register.*

line):

```
⟩ E4     TR     Text="Check
```

You should now change the format of the numbers. At the start, SuperCalc displays numbers as they "best fit" in their cells. If you type a number with a decimal point (. — *not* a comma as used in Europe), SuperCalc displays the number as a decimal number; however, other whole numbers are displayed as integers, and columns of numbers may not appear to line up, unless you choose to line them up.

You can decide to display numbers, including decimals, as integers with the "I(nteger)" format. You can also display numbers as decimals (whole numbers have two zeros following the decimal point) with the *$* (dollars and cents) format. *All numbers are stored in memory exactly as you typed them, or as they are calculated, regardless of display format.* They are rounded into integers for display only, and only for the "I(nteger)" format.

Since this spreadsheet uses dollars and cents, you should use the dollars and cents format. Type the following Format command: */FG$↩* (slash, *F*, followed by *G* for "G(lobal)," a dollar sign, and RETURN). All integers on the spreadsheet instantly turn into decimal numbers.

One problem: the A column is a column of integer numbers representing check numbers. Move the cursor to column A, and type the following command: */FC↩I↩* (slash, *F*, followed by *C* for "C(olumn)," followed by a RETURN for the current column, followed by an *I* for "I(nteger)," followed by RETURN).

Your spreadsheet should look like the one in figure 4, with integer numbers in column A, decimal numbers in columns E, F and G, and right-justified labels in cells E4, F4 and G4. The balance formula should be working.

Saving the Spreadsheet

If your formula for cell G7 is working, and you have properly formatted your columns and cells, you should save this version of the spreadsheet in a file that you can change or delete later. Type */S* for the Save command, and type a filename as shown (you can include with the filename a drive prefix to specify a disk drive other than the current disk drive):

```
Enter File Name (or ⟨RETURN⟩ for directory)
7⟩/Save,REGISTER↩
```

(You can also press RETURN, without a filename, for a directory display of SuperCalc spreadsheet files on disk, or all files on disk. If you have *User's Guide* #2, see page 24, for a description of this feature, available also with the Load command.)

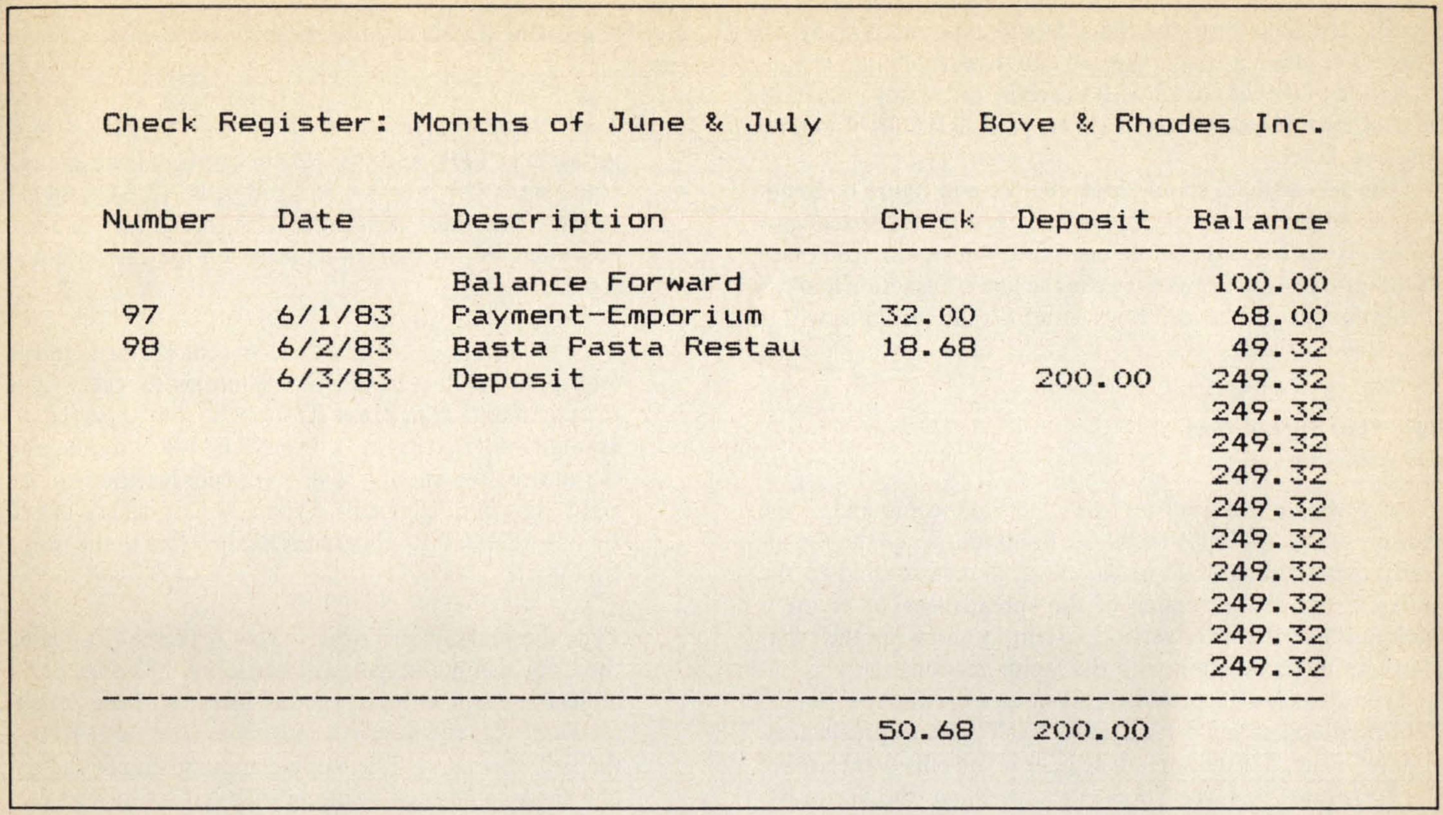
After typing a filename and pressing RETURN, SuperCalc displays the following:

```
A(ll), V(alues), or P(art)?
```

Type *A* for all, and SuperCalc saves the entire spreadsheet, with formulas, to the file (we use the name REGISTER). SuperCalc returns control to you after the save, with the spreadsheet on your screen (unchanged by the save operation).

Replicating a Formula

The formula in G7 is Previous Balance + New Deposit -



```
Check Register: Months of June & July          Bove & Rhodes Inc.

Number    Date       Description              Check   Deposit   Balance
------------------------------------------------------------------------
                     Balance Forward                            100.00
 97       6/1/83     Payment-Emporium         32.00              68.00
 98       6/2/83     Basta Pasta Restau       18.68              49.32
          6/3/83     Deposit                          200.00    249.32
                                                                249.32
                                                                249.32
                                                                249.32
                                                                249.32
                                                                249.32
                                                                249.32
                                                                249.32
                                                                249.32
                                                                249.32
------------------------------------------------------------------------
                                              50.68   200.00
```

Figure 6. *Check Register with check and deposit totals. Use the Global command with the "B(order)" option to make the outer row and column labels disappear from the screen and from printouts.*

New Check. By using the Replicate command, this formula can be replicated to the cells below it, and its coordinates can be changed to be accurate for each cell.

Move your cursor to cell G7. Type */R* and press RETURN to replicate G7. Then press your ESC key, and move the cursor to cell G8. Type a colon (:), and move the cursor down column G to cell G18. Now press RETURN.

The column should fill up with the number 62.00 (or whatever number you calculated for cell G7). Yet, if you type new transactions (checks or deposits) in the spaces provided, this balance amount changes to reflect each new transaction. An automatic check register and calculator!

Try it: move over to cell A8 and type check number 98. If you move vertically to row 8 and then horizontally to cell A8, the data entry process will be smoother. Type in a date as a label (preceded by a quotation mark), a description, and another check amount. Then move to B9 (skipping A9) and type a date and a deposit. See figure 5 for an example.

The formula for cell G7 is shown on the message line as G6+F7-E7. As you move the cursor down column G, the formula's coordinates change for each cell, but the formula itself is the same.

Summing Columns

To create column totals, move your cursor down to cell A19. Type the command */R5* and RETURN, then press RETURN again to replicate row 5 (the line of dashes) to row 19.

Move your cursor to cell E20, and type the following formula:

```
1)sum(e7:e18)↩
```

This formula adds all values from E7 to E18, and thereby sums all check amounts. To copy this formula to cell F20, you can use the Replicate command, or you can use the Copy command, which copies one cell. Type */C*, followed by *C20* and RETURN, followed by *F20* and RETURN. The formula is copied and the coordinates changed in relation to the new cell's coordinates.

Editing a Cell

To learn the handy cell editing feature, move your cursor to the A1 cell and type the */E* command. SuperCalc displays:

```
From? Enter cell
26)/Edit(into current cell),
```

Press RETURN for the current cell (A1), and SuperCalc displays what you typed for cell A1 on the command line, with the cursor at the end of the entry:

```
31)"Check Register: Month of June
```

You edit the line by typing over the characters to change. Move your cursor with your left and right cursor control keys (↑*S* for left and ↑*D* for right). Use the up-arrow key (↑*E*) to insert a space between characters, and use the down-arrow key (↑*X*) to delete a character or space.

Move the cursor to the end of Month, type an up-arrow key (or ↑*E*) to insert a space, type an *s* to turn Month into Months, and then move the cursor with the right-arrow key (↑*D*) to the end of June to add *& July*. Finally, press RETURN to enter the new label.

The spreadsheet should look complete in figure 6. To add entries to the spreadsheet, you can type in the spaces provided. To increase the space provided, you can use the Insert command to insert rows between the last transaction and row 17. If you insert between rows 8 and 17, the formulas will be adjusted correctly.

Save Your Spreadsheet and a Backup Copy

Save your check register with the Save command. Type the command */S*, followed by the filename. Since the file already exists, SuperCalc gives you a chance to change the name of the newer version of the spreadsheet, or create a backup file of the older version, or simply overwrite the existing version (replacing it with the newer version).

Type *B* to save the previous version as a backup file. SuperCalc then asks if you want to save "A(ll" of the spreadsheet, or only "P(art" of the spreadsheet, or if you only want to save the "V(alues)" of the spreadsheet without the formulas. Type *A* to save the entire spreadsheet.

The spreadsheet can be used for two months, with Save operations after each transaction. Use the */S* command and select the *B* (for Backup) option so that the previous day's spreadsheet is also saved.

At the end of the months of June and July, Load the June & July spreadsheet. You can use this editing function to change label A1 to August or August & September.

Use the Blank command with the range A7:F18 to blank-out all cells under the Number, Date, Description, Check and Deposit headings. These cells contain values or labels; be sure that you do not blank-out cells containing formulas.

Finally, provide a new balance forward amount, and the spreadsheet is ready for new checks and deposits. Be sure to save the new month's spreadsheet in a new file. Also, be sure to make archival copies of the files onto different disks (in case your working disks fail). The CP/M PIP command is adequate for copying SuperCalc worksheet files.

Ratios and Averages: A Sales Forecast by Season

Our next example has simple calculations, yet it shows basic spreadsheet creation and editing techniques, the use of ratios and the Average function, and extensive use of the Copy and Replicate commands.

Many businesses operate in a seasonal way. For example, book sales are high before Christmas and before summer, and moderate to low during other the other seasons. A book publisher may build from this simple model a spreadsheet to forecast book sales over the next five years.

Figure 7 shows the spreadsheet headings. Here is each step for creating it (you can use the same steps to create your own):

1. Type *"Sales Forecast 1983-1987* in cell A1, *"B & R Publishing* in cell D1, and *"4/1/83* in cell G1. Type enough equal signs (*"= = = = = =* ...) in cells A2, D2 and G2 to underline these headings. (Use the */E* command to edit each cell to shorten or lengthen the line of equal signs.)

2. Type the headings *"Q1*, *"Q2* etc. in cells B7 through E7, and type *"Yr. Total* in cell G7. While on that row, type the command */FR*, press RETURN, and type the *TR* option (for "TextRight"). Press RETURN again, and the entire row should have its labels justified to the right margin of each cell. Type a line of dashes in cell B8 (*"—-* ...) and Replicate the dashed line to the range C8:G8.

3. Type the years as row titles in row A (A9:A13). Using the Copy command, you will now copy the entire block of headings and titles to another location. Type */C*, enter *A4:G13* as the range to copy from, and enter *A16* as the cell to copy to. The Copy command copies the entire block cell by cell, starting with cell A4 to A16, etc.

4. Use the Copy command again to copy the entire block, A4:G13 to another destination: cell A30.

5. Change the labels in cells D16 and G19 to "Yearly Index" and "Grth Rate", and edit the dash line in cell D17 to be one dash shorter. Type the */FR* command on row 19, press RETURN to select the current row (19), and type *TR* and RETURN to make the labels right-justified within their cells.

6. Change the label in cell D30 to "Sales Projection" and add a few dashes to the line in cell D31. Move to row 33 and type the */FR* command, press RETURN, and type *TR* to right-justify the text in the row. Change the labels in cells A35 through A39 to the years 1983 through 1987, and type the label "Average" into cell A27.

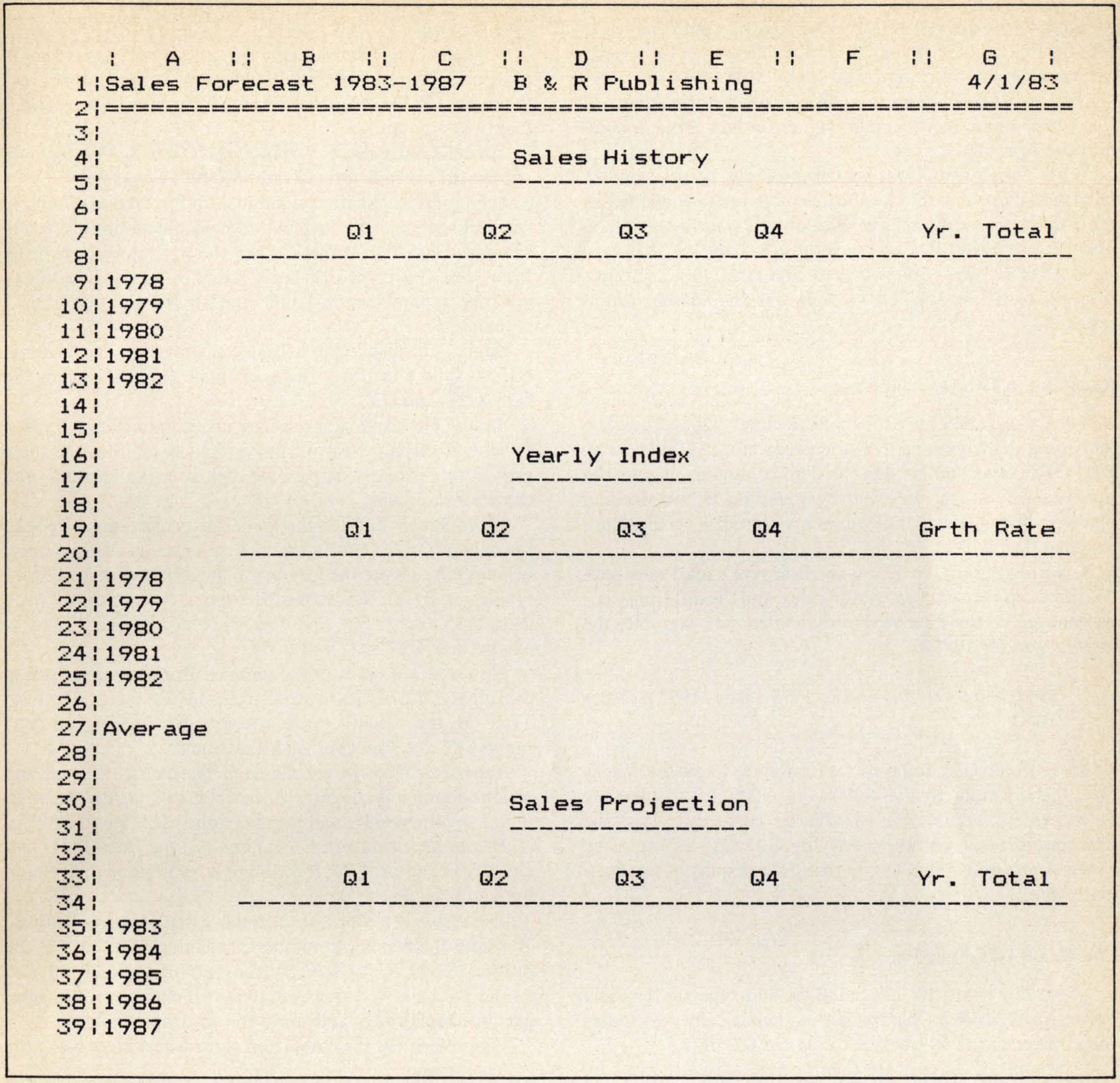
Efficient use of a computer for this task comes with practice at quick editing and copying. The better you are at setting up working models, the faster you can be in achieving results with them.

Replicating With Manual Adjustment

Enter values for the sales history portion of the spreadsheet (the block B9:E13). Enter reasonable sales figures to reflect seasonal business — high sales in Q2 and Q4, lower sales in Q1 and Q3.

Move to cell G9 and type the formula *SUM(B9:E9)*. Use the Replicate command to replicate the formula to the cell range G10:G13.

To enter the yearly index formulas, move your cursor to cell B21. The basic formula for each quarterly index is:



```
   ¦   A    ¦¦  B    ¦¦  C    ¦¦  D    ¦¦  E    ¦¦  F    ¦¦  G    ¦
  1¦Sales Forecast 1983-1987   B & R Publishing               4/1/83
  2¦================================================================
  3¦
  4¦                           Sales History
  5¦                           -------------
  6¦
  7¦                Q1       Q2       Q3       Q4         Yr. Total
  8¦         -------------------------------------------------------
  9¦1978
 10¦1979
 11¦1980
 12¦1981
 13¦1982
 14¦
 15¦
 16¦                           Yearly Index
 17¦                           ------------
 18¦
 19¦                Q1       Q2       Q3       Q4         Grth Rate
 20¦         -------------------------------------------------------
 21¦1978
 22¦1979
 23¦1980
 24¦1981
 25¦1982
 26¦
 27¦Average
 28¦
 29¦
 30¦                           Sales Projection
 31¦                           ----------------
 32¦
 33¦                Q1       Q2       Q3       Q4         Yr. Total
 34¦         -------------------------------------------------------
 35¦1983
 36¦1984
 37¦1985
 38¦1986
 39¦1987
```

Figure 7. *Headings and labels for the sales forecast model.*

Quarterly Index = (Quarterly Sales)/Total Sales

The formula for cell B21 is *B9/G9.* Using the Replicate command, you can copy this formula to the other cells on row 21. However, you must use an option in the Replicate command to avoid changing one of the coordinates — the Yearly Total (G9).

Move to cell B21, type */R* followed by RETURN, and type the range as *c21:e21*; however, *do not* press RETURN yet!

First, type a comma (,). The comma defines the destination range as C21:E21, and SuperCalc displays more options:

```
N(o Adjust), A(sk for Adjust), V(alues)
```

Type *A* to "Ask for Adjustment" of the formula's cell coordinates. SuperCalc displays the first cell coordinate in the formula:

```
Source cell B21. Adjust B9 (Y or N)?
```

Type *Y* to adjust this coordinate (it will change to C9, D9 and E9). SuperCalc then displays the next coordinate:

```
Source cell B21. Adjust G9 (Y or N)?
```

Type *N* to leave G9 as it is (otherwise, it would change to H9, I9, etc., causing errors). SuperCalc replicates the formula changing one coordinate and leaving the other the same. Using this "Adjust" feature, you can ask for adjustment of every coordinate in a formula, no matter how large or complex the formula.

With the formula replicated properly across row 21 (B21:E21), use a simple Replicate command to copy the formula to the rest of the cells with automatic coordinate adjustment. Move to cell B21 and type the command */R*, then type *b21:e21* as the range to copy from, and *b22:b25* as the range to copy to. Press RETURN, and you should see instant results.

Rounding to Two Decimal Places

The index values have too many decimal places. To adjust that, type the command */FE*, the range *b21:e25* followed by RETURN, and a dollar sign (*$*). This command changes the way the values are displayed by rounding to two decimal places; however, the actual values are stored as accurate unrounded decimals.

The growth rate formula is the first year's total sales subtracted from the next year's total sales, and the difference is a percentage of the first year's total sales. For example, the growth rate for 1979 is:

> (Grth Rate 1979) = (Sales 1979 - Sales 1978)/(Sales 1978)

Move to cell G22 and type the formula *(g10-g9)/g9*. Replicate this formula to the cell range G23:G25 (SuperCalc automatically adjusts the coordinates correctly). Type the */FE* command as before, specifying G22:G25 as the entry range, and a dollar sign (*$*), to round the display to two decimal places.

The AVERAGE Function

Move down to row 27, cell B27, and type the formula *average(b21:b25)#+*. SuperCalc automatically calculates the average of the numbers in the range B21:B25.

From cell B27, replicate this formula across the row by typing */R*, RETURN, and the range *c27:e27*. Replicate the formula again to cell G27. SuperCalc automatically adjusts the coordinates correctly.

From row 27, type the command */FR*, followed by RETURN, followed by a dollar sign (*$*) and RETURN, to round the displayed numbers to two decimal places.

More Replicating and Adjusting

For the sales projection area, the formulas for each quarter's projection is based on the quarter's average index, and each total is based on the overall average growth rate. You can change these formulas to be more precise, calculating projections using the actual year's quarterly index values and growth rate.

For cell G35, type the formula *g13*(1+g27)*. This formula multiplies the total sales of 1982 by the average growth rate to project 1983's total sales. The model derives its projections from this amount.

For cell G36, type the formula *g35*(1+g27)*. This formula applies the average growth rate to 1983's total sales.

Using the Replicate command with the "Adjust" option, you can replicate the formula without changing the second coordinate (G27) — the one for the average growth rate. From cell G36 type */R*, press RETURN, type the range *g37:g39*, type a comma, and type an *A* for the "Adjust" option.

SuperCalc displays the first coordinate in the formula (G35). Type *Y* to adjust this coordinate (so it can change to G36, G37, and G38).

SuperCalc displays the second coordinate (G27). Type *N* to leave it the same for all three replicas (so that the same growth rate is used). SuperCalc replicates the formulas and calculates the cells.

Move to cell B35 for the quarterly projections. Type the formula *g35*b27* (1983 total sales times the average quarterly index). Replicate the formula to the cell range C36:E36 by typing */R*, RETURN, *c36:e36*, comma, *A* for "adjust", *N* to leave G35 unchanged, and *Y* to adjust coordinate B27 (so it can become C27, D27 and E27).

Now you are ready for a complex Replication adjusting coordinates to replicate to all cells in the block B36:E39. Type */R*, the "from" range *b35:e35*, RETURN, the "to" range *b36:b39*, a comma, and *A* to adjust.

SuperCalc displays the message Replicating...B to tell you which column it is copying to. Since it is copying to column B, type *Y* to adjust G35, and *N* to not adjust B27 (to use B27 in all B-column formulas). For column C, type *Y* to adjust G35, and *N* to not adjust C27; for column D, type *Y* and then *N*, and for column E do the same.

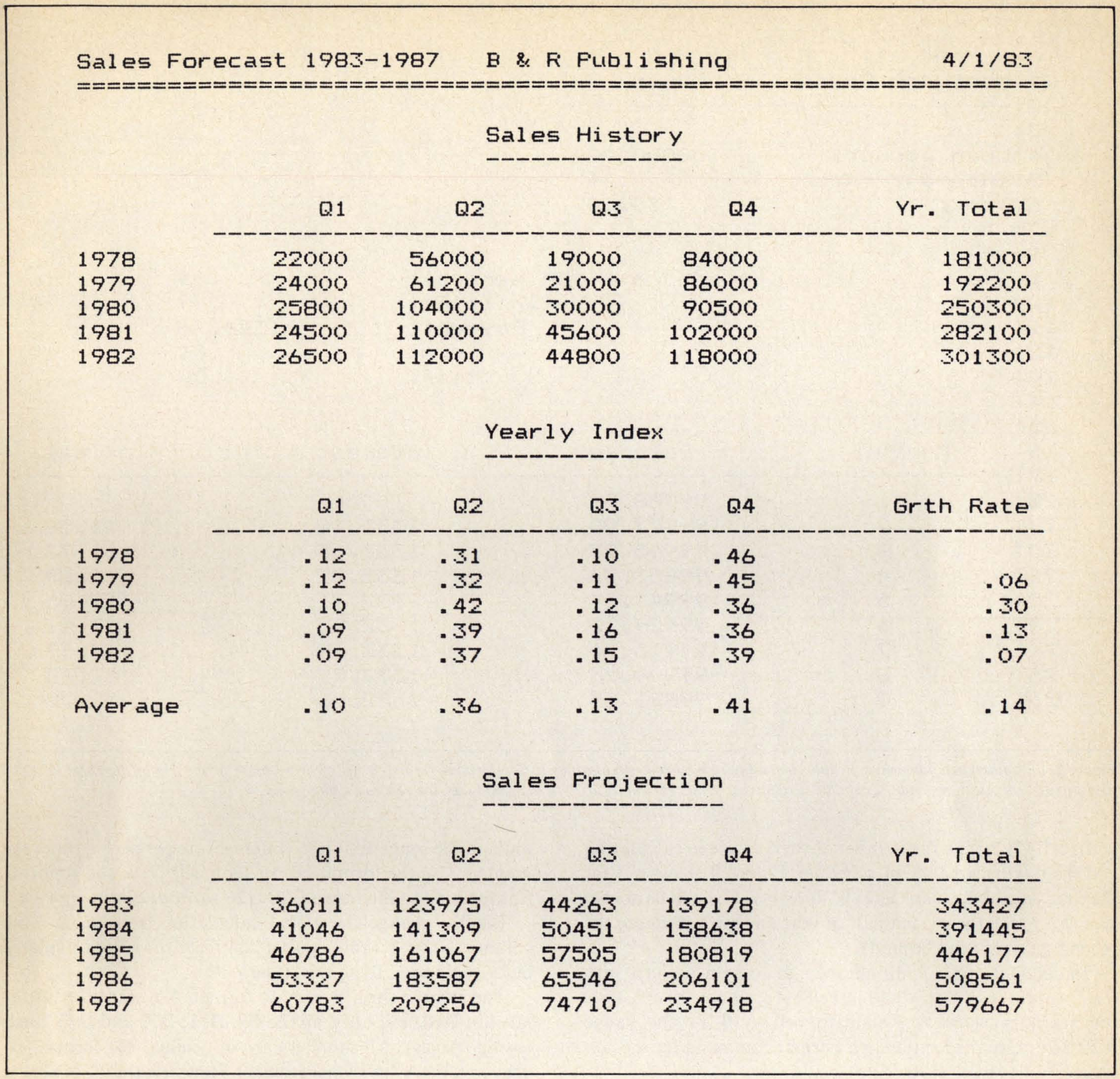
The result is a block of formulas calculating projections. To round the displayed numbers into integers, type */FE*, the range *b35:g39*, RETURN, an *I* for the "Integer" format, and a final RETURN. SuperCalc rounds the numbers into integers for display only, and stores the accurate amounts.

Figure 8 shows the completed spreadsheet after using the */GB* command.

You can produce a crude graph of your sales history by multiplying all values in the Grth Rate column by 100, widening the column to at least 50 characters, and using the asterisk format (*/FC↩* command followed by *) for the column. The number of asterisks displayed in each cell equals the integer value of each cell (this is why you must widen the cell's width).

"What If" Calculations: A Mortgage Schedule

SuperCalc shows its powers of recalculation in this example of a mortgage payment schedule. To examine all the possible repayment plans and interest rates, you use SuperCalc



```
Sales Forecast 1983-1987   B & R Publishing               4/1/83
==================================================================

                            Sales History
                            -------------

                Q1        Q2        Q3        Q4         Yr. Total
          --------------------------------------------------------
1978         22000     56000     19000     84000            181000
1979         24000     61200     21000     86000            192200
1980         25800    104000     30000     90500            250300
1981         24500    110000     45600    102000            282100
1982         26500    112000     44800    118000            301300

                            Yearly Index
                            ------------

                Q1        Q2        Q3        Q4         Grth Rate
          --------------------------------------------------------
1978           .12       .31       .10       .46
1979           .12       .32       .11       .45               .06
1980           .10       .42       .12       .36               .30
1981           .09       .39       .16       .36               .13
1982           .09       .37       .15       .39               .07

Average        .10       .36       .13       .41               .14

                            Sales Projection
                            ----------------

                Q1        Q2        Q3        Q4         Yr. Total
          --------------------------------------------------------
1983         36011    123975     44263    139178            343427
1984         41046    141309     50451    158638            391445
1985         46786    161067     57505    180819            446177
1986         53327    183587     65546    206101            508561
1987         60783    209256     74710    234918            579667
```

Figure 8. *Completed sales forecast model.*

to answer "what if" questions by varying a few numbers in a table and watching the results appear on the screen.

A simple mortgage schedule is shown in Figure 9. The "variables" are kept at the top of the spreadsheet, where they can be changed easily. These "variables" are the loan amount, the interest rate, and the term of the loan (number of years). These values are in the cells C4, C5 and C6. They are the only values you have to supply to calculate the rest of the schedule.

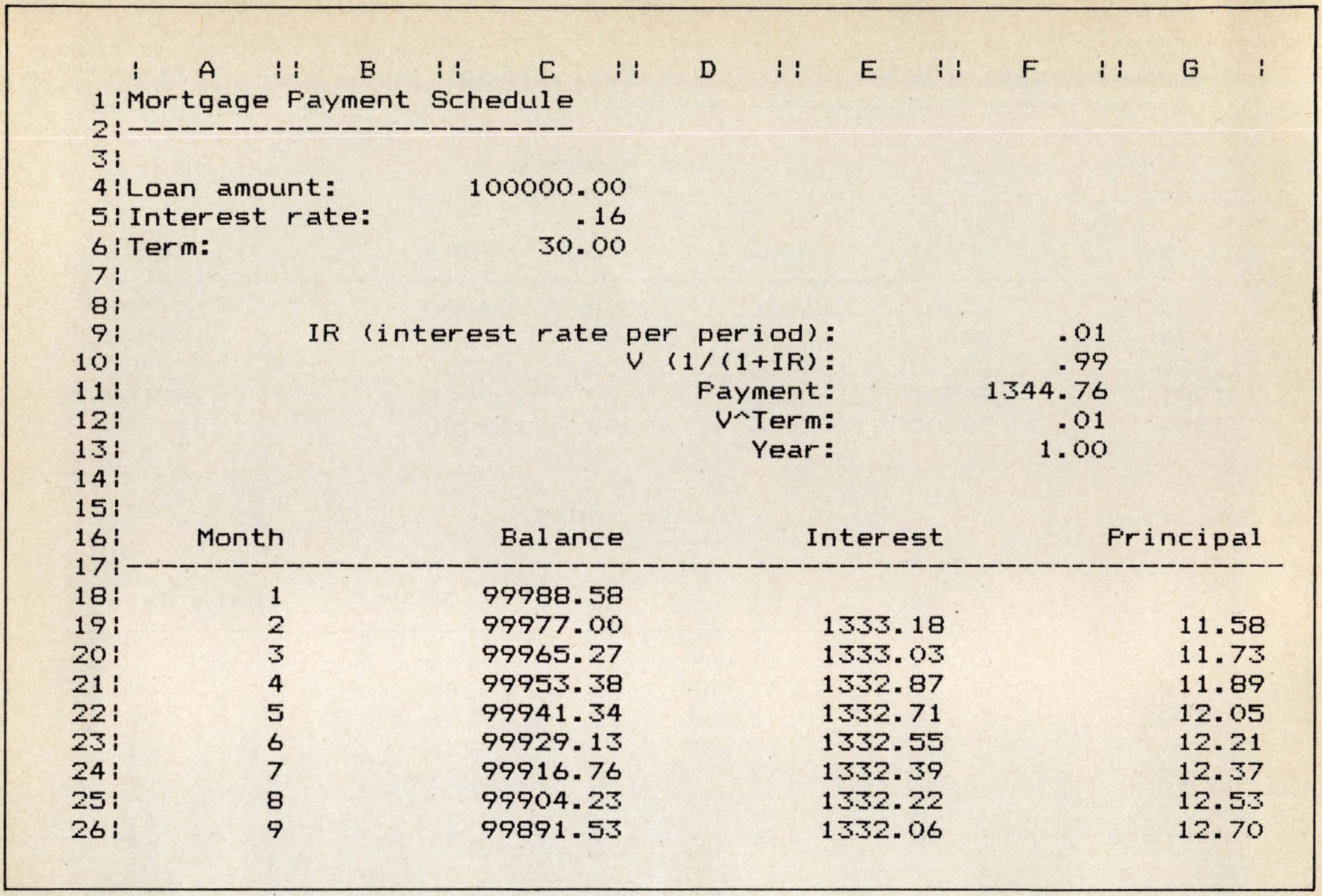
To build the schedule, start with some calculations that make the schedule itself easy to calculate. In other words, to save time, place these formulas in cells so that the cells can be used in other formulas to calculate the schedule:

Cell F9: interest rate per month (IR)
  Interest Rate / 12
  Use the formula *C5/12.*

Cell F10: V
  V = 1 / (1 + IR)
  Use the formula *1/(1+F9).*

Cell F11: payment per month.
  Payment = Loan Amount x IR / (1 - (F10 ↑ no. of months))
  Use the formula *C4*F9/(1-(F10↑(12*C6)))*

Cell F12: V ↑ term
  V ↑ (number of months)
  Use the formula *F10↑(C6*12).*



```
   |   A    ||   B    ||   C    ||   D    ||   E    ||   F    ||   G    |
  1|Mortgage Payment Schedule
  2|-------------------------
  3|
  4|Loan amount:        100000.00
  5|Interest rate:            .16
  6|Term:                   30.00
  7|
  8|
  9|         IR (interest rate per period):             .01
 10|                            V (1/(1+IR):            .99
 11|                                Payment:        1344.76
 12|                                 V^Term:            .01
 13|                                   Year:           1.00
 14|
 15|
 16|    Month            Balance           Interest          Principal
 17|-------------------------------------------------------------------
 18|        1           99988.58
 19|        2           99977.00            1333.18              11.58
 20|        3           99965.27            1333.03              11.73
 21|        4           99953.38            1332.87              11.89
 22|        5           99941.34            1332.71              12.05
 23|        6           99929.13            1332.55              12.21
 24|        7           99916.76            1332.39              12.37
 25|        8           99904.23            1332.22              12.53
 26|        9           99891.53            1332.06              12.70
```

Figure 9. *A mortgage payment schedule, featuring row-by-row calculations (before column totals), a list of "variables" at the top of the sheet, a set of preliminary calculations, and the capability to answer "what if" questions with loan amounts, interest rates, and terms.*

In cell C12, type the number of years to use in calculating and displaying a payment schedule. Figure 9 shows a *1* for the first year of the loan (term is 30 years). You will be able to type *1.5* to see the second half of year one, or *1.25* to see the second quarter (and beyond).

To calculate the month numbers in column A, start with the formula *1+(12*(F13-1))* in cell A18. For cell A19, type the formula *A18+1*. Replicate cell A19 to the range A20:A26. Use the *I* option in a Format command for the column.

For the first monthly balance (C18), type the following formula:

```
C4/(F10↑A18)*((F10↑A18)-F12)/(1-F12)
```

The formula is:

Balance = Loan / (V↑month)
x (V↑month - V↑term) / (1 - V↑term)

Replicate this formula from C18 to the range C19:C26, adjusting only two coordinates in the formula: the A18 coordinate (used twice). This changes to A19, A20, etc., but the other coordinates *must not* change.

Use the formula *F9*C18* in cell E19, and replicate the formula to the range E20:E26 (interest rate per period times the month). Use the formula *F11-E19* in cell G19, and replicate this formula to the range G20:G26 *without adjusting* F11.

Finally, change the global calculation from column-by-column to row-by-row, so that complete rows are calculated before columns. To do this, type */GR*.

The sample sheet is easy to format. We started with the label in D10, and lined up the B9, D11, D12 and D13 labels (using spaces). All numbers are in decimal (*$*) format, except column A (in *I*nteger format). Rows 16 and 17 are set to the *TextRight* format.

We use this model to calculate mortgages, auto loans, business loans, etc.

Automatic Operations

SuperCalc lets you create *command* files holding the commands you would type on the SuperCalc command line. We use this feature to do automatic printing of a set of spreadsheets without having to type the individual commands to load and print them. These command files usually have an ".XQT" extension.

There are two ways to execute a command file. You can use the */X* command within SuperCalc, or you can specify the name of the command file with the SuperCalc invocation command.

For example, suppose you have three spreadsheets to print whenever you prepare a special report: BALANCE.CAL, INCOME.CAL and CASHFLOW.CAL. Start by defining the exact SuperCalc commands to load and print each spreadsheet, and put each spreadsheet's commands in a separate file. These files can be created with any text editor.

Start with the command file BALANCE.XQT, used to print the BALANCE.CAL spreadsheet:

```
/Lbalance.cal,A            (Load BALANCE.CAL.)
/ODAP                (Output display to printer.)
/QY              (Quit and prepare for next one.)
```

Create INCOME.XQT and CASHFLOW.XQT using the above example as a template (substitute appropriate filename for *balance.cal* above).

Create another file with the ".SUB" extension to use with the CP/M SUBMIT command (called REPORT.SUB for this example):

```
SC BALANCE↵
SC INCOME↵
SC CASHFLOW↵
```

The *SC* command will accept as an argument the name of the ".XQT" file, and execute the commands found in the file. Each *SC* command is executed in "batch" mode by SUBMIT:

```
A)SUBMIT REPORT↵
```

The SUBMIT command executes commands found in REPORT.SUB. These *SC* commands start SuperCalc and execute the commands found in each ".XQT" file. In this manner you can print all three spreadsheets by typing one SUBMIT command.

End of Lesson

SuperCalc users develop habitual ways of creating spreadsheets. We have developed our own techniques, shown in this article, and we encourage feedback from other SuperCalc users who perhaps have developed better ways of doing things.

There are several spreadsheet models commercially available to solve specific problems, and there are several books for SuperCalc users with general-purpose models as well as specific ones. We will be reviewing commercially-available models and books, and providing more application notes on SuperCalc. In addition, in this issue you'll find a brief summary of SuperCalc commands.

It is the nature of a product like SuperCalc to provide different ways of solving problems. We appreciate the fact that spreadsheet models can be very different yet achieve similar results — this is the "personal" nature of personal computing.

3 NEW GUIDES TO PROGRAMMING UNDER CP/M—ALL FROM RESTON PUBLISHING COMPANY, YOUR SOURCE FOR COMPUTER BOOKS

A Programmer's Notebook: Utilities for CP/M-80
David Cortesi

This book gives you the concepts and techniques needed to create professional quality programs in 8080 Assembly Language. You'll be able to build an impressive software library with the many ready-to-run programs included in the book — not available in other books. At the same time, you'll learn how to develop and implement custom designed programs for your own specific needs. Every step is thoroughly examined and explained, from concept to the final working program. Paperback.

System Programming Under CP/M-80
Lawrence Hughes

Lawrence Hughes provides you with a thorough and comprehensive introduction to 8080/8085 Assembly Language programming, specifically under the CP/M operating system. It's filled with new and original source programs designed to help you write your own programs. This book contains many usable utility programs for CP/M along with a detailed analysis of each program. Some of the topics included are: using BIOS, BIOS design and system generation, internal operation of CP/M, modular assembly language development, and more. Paperback.

A User-Friendly Guide to CP/M
James T. Perry/Robert F. McJunkins

Finally, an easy-to-read, easy-to-follow guide to using the CP/M operating system. It's ideal for first-time users — taking them from an introduction to CP/M to creating new dBASE commands. In between, you'll find topics such as built-in commands, copying a large number of files, system files, user areas, copying selected parts of files, introduction to dBASE II, and altering the data base contents. Paperback.

Please send me . . .
____ copies of R5641-7, A Programmer's Notebook, $16.95
____ copies of R5643-3, A Programmer's Notebook disk/book package, $34.95
____ copies of R7456-8, System Programming Under CP/M-80, $15.95
____ copies of R8117-5, User-Friendly Guide to CP/M, $14.95
____ free copy of computer books and software catalog.

SAVE: If check or money order accompanies order, publisher will pay postage. Please enclose your state sales tax.

Send order to: RESTON PUBLISHING COMPANY, 11480 Sunset Hills Rd. Reston, Va. 22090

Name __________
Address __________
City ______ State ______ Zip ______

Circle reader service number 12

Letters to the Editors...

Osborne Observations

I just received my second copy of *User's Guide*, and for the first time since buying the Osborne Computer I have been able to use it for something other than chess.

I am unable to move about freely (emphysema). Do you know of anyone that I might correspond with that would take the time to teach me how to use Osborne (including dBase II) software, and possibly some basic programing. I can't pay for the instructions, but I can pay for the postage. Help if you can.

Paul Swift
250 Baldwin Ave., 202
San Mateo, CA 94401

Public Domain Info

Hi. Found these on the Usenet 4 and thought you might be interested:

From: Rick Conn

The New York Amateur Computer Club, Inc, has published a series of documentation catalogs of the public domain software. There are 7 catalogs so far, each containing over 200 pages of documentation on public domain software. This documentation is almost exclusively copied from the documentation files on the public domain disks themselves.

I skimmed through all 7 catalogs this afternoon, and they seem to be quite worthwhile to get. As I see it, they provide three main services:

1. A good overview of the public domain software in SIG/M and CP/MUG; more than just listing the names of the files and programs on these disks, the documentation is included, so you can scan the documentation to see what you will get before you acquire the disks.

2. A *hard copy* of the documentation; once you obtain a disk of software, you usually want to print out the documentation anyway; these catalogs are in good, letter-quality print and they contain tables of contents.

3. A consolidation of the information on public domain software, provided at a reasonable cost (the NYACC sells them at just slightly above cost, and the profit is used to fund club activities.

Several people have been expressing concerns about the ZCPR2 and SYSLIB documentation. Among other things, the last three catalogs contain *all* of the ZCPR2 and SYSLIB documentation.

The catalogs contain the following information:

Book 1 (204 pages): CP/M 1 to 49; SIG/M to 18.

Book 2 (214 pages): CP/M 50 to 54, 78, 79 (CP/M 55 to 77 are re-issues of SIG/M 1, 2, 11, 4-10, 13-25 resp); SIG/M 19 to 42.

Book 3 (217 pages): SIG/M 43 to 60.

Book 4 (210 pages): CP/M 80 to 84; SIG/M 61 to 76.

Book 5 and Book 6 (400+ pages); SIG/M 77 to 106 (ZCPR1 is in SIG/M 77 and ZCPR2 is in SIG/M 98 to 108; SYSLIB IS IN SIG/M 88 to 90, 106 to 108).

Book 7 (200+ pages): SIG/M 107 (SYSLIB documentation).

For those interested in ZCPR2 and SYSLIB documentation, Books 5 to 7 contain good hard copies of all of the documentation.

Each book costs $10 (including shipping) in North America and $15 (via airmail) overseas. All orders must be prepaid in US Funds. The address to write to for ordering the books is:

New York Amateur Computer
Club, Inc
PO Box 106
Church St Station
New York, NY 10008

Note: the NYACC is a non-profit organization. By purchasing their materials, you are contributing to their club's activity funds as well.

One last note about the NYACC; those interested in their catalogs can phone Susan Perricone at (212) 243-0325 for more info. I believe that details such as unlimited distribution and copy permission within a computer club or company may be worked out. Reselling was also mentioned.

John Gilmore
San Francisco, CA

Thanks to John, Rick and USENET.

Congratulations on your first two fine issues of *User's Guide*.

I have communicated by telephone to some of the RCPMs with my Smartmodem using Modem7. The problem I have found is that most of the good programs in the public domain such as MODEM765.ASM assembler source occupy almost 100K with the documentation and macro library. Using a modem at 300 baud (30 bits per second) the time it takes is almost an hour. It is easier to get them through the User Group for the low cost of copying.

I have the complete CP/M User's Group (CPMUG) public domain programs downloaded to Intertec Superbrain/Compustar quad density disks only. The cost of copying is $12 per volume.

Jack Wittmann
T.E.C.
7870 Becker Rd.
St. Louis, MO 63129

Overly Cautious

As a ten-hour-a-day WordStar user

(I'm writing a book on an Osborne), I greatly admired your concise introduction to WordStar in issue No. 2.

However, I think you may have been overly cautious in warning users to exit to CP/M before changing data disks in the B drive. Several fellow WordStar users and I have found that if we are logged onto drive B (with WordStar in drive A), we can simply switch data disks while at the No-File menu, then use the *L* command (change logged drive) to re-log onto drive B. The new disk B directory then pops onscreen. This is obviously much quicker than having to reload WordStar and reset margins, etc., every time one changes data disks.

I have read several implicit warnings, including yours, that a method such as the above can result in data loss. But in six months of using it I have yet to encounter — or hear of — a problem.

Janet Goldenberg
San Francisco, CA

Thanks, Janet. You may be right, for we at times perform the same operation. However, once, while using an earlier version of WordStar, we tried this and upon saving the text file, the system displayed the DISK IS R/O *error message associated with trying to write to a disk that has not been logged in. The circumstances were not fully understood at the time; perhaps the glitch occurred for another reason...*

On page 30, Vol 2 of *User's Guide* you issue a warning not to change diskettes while in WordStar.

I commonly change ".TXT" file disks (Drive B) from the No File menu by typing *L* in the menu, then *B:*, then change disks and type RETURN.

It appears that WordStar issues a Control-*C* by this method. The directory changes to the new disk, and I have noticed no apparent problems.

We also run DataStar under CP/M on an Osborne 1 single density machine to store about 4,000 700-byte records over several disks. Since DataStar requires a little over half-disk space to perform file maintenance, we were constantly running out of disk space during file maintenance, requiring a Supersort run to move part of the file to another disk.

To solve this annoyance we now specify Drive A: for the Batch file during file maintenance, and replace the system disk with a formatted and "Sysgened" blank disk before typing RETURN.

The system hangs upon pressing ESCAPE after file maintenance is complete, and must be reset, but no apparent directory damage is done as all files read correctly thereafter, and it allows us to use almost full disk space.

One problem we haven't solved using DataStar is changing data disks during a Scan Mask Sort. Once a month we perform a search of the aforementioned 4000 records spanning some 40 disks for clients needing attention that month. Each time a disk is changed Datastar must be exited, which resets the Scan Mask, requiring the Scan Mask to be re-edited for each disk or some 40 times.

A routine or patch that could be called to reset disk B: without having to exit the Scan Mask would be a true and genuine blessing.

Don M. McLaughlin
Brandon, FLA

Article Reprints

I am a new rookie in the computer sales business, and I have found your magazine's tutorials very helpful. I am writing to inquire about reprints. Are they available? What is the cost and delivery time?

If reprints aren't available yet, how about your permission to reproduce the tutorials for personal use and for clients?

Thanks for taking on the task of making CP/M software understandable!! Take Care.

Vicki C. George
Topeka, KS

Article reprints are available, price determined by type of reprint, size of article, and number of copies. Let us

What's wrong with CP/M®?

Hard to operate • Too many commands and options • Too many ways perfectly good programs can fail • Not enough specifically helpful messages...

It needs OKARA™!

In 1981-82, many vendors marketed CP/M user interfaces (also called front ends, shells or menu drivers).

All of them (save one) were mastodons: slow, clumsy, inflexible, disk-hungry, hard to install, and overpriced. Their buyers usually tried them, sighed, shelved them, and forgot them.

OKARA, on the other hand, is a gazelle. Once you have tried OKARA, you will never willingly go back to bare CP/M. Why? Because OKARA is

- ☐ **Incredibly fast:** It operates at memory speed.
- ☐ **Automatic** and nearly invisible in operation.
- ☐ **Flexible** and programmable — You decide how it communicates and just how much control it exercises.
- ☐ **Immediately usable** by the most casual beginner, yet powerful enough for the most demanding OEM.
- ☐ In some ways **more capable than CP/M 3.0,** yet it runs on CP/M 2.2.
- ☐ **Field proven** in many real-world applications.
- ☐ **Low priced:** $39.95.

This is the same first-quality software package we used to sell for $150. We now proudly join the industry-wide trend toward reasonable software prices.

OKARA runs on any standard CP/M 2.2 system. We distribute it in CP/M-standard 8" SSSD, Northstar SSDD, and Osborne SD formats.

Order today. We're sure you'll be delighted, and we have stacks of testimonial letters to prove it.
We are...

KIAI systems
180 Grand Ave.
Suite 900
Oakland
CA 94612
(415) 654-8671

California residents add sales tax. Add $3.00 for COD's, $10.00 for net 30. CP/M is a trademark of Digital Research.

Circle reader service number 9

know what you want once you have read the copyright and subscription information on our table of contents page.

Spring Has Come

Your article "Downloading From RCPM Systems" was most interesting and educational. I am using a vintage TRS-80 Model I with the Omikron Mappers I and II, 48K, two 5" and two 8" floppy disk drives to mostly drive the Magic Wand software. The CP/M format until now has been mostly aggravation, because of the flakiness of the Model I and the total lack of utilities that would let me work directly on the disks to repair or rebuild their contents. The ED and DDT programs were in 8080 assembly language and it's difficult to learn to translate to Z80.

But now not only do I see the end of the tunnel with your article, but the sun has come out and spring has come.

However, a question: how do I go about getting a copy of the MODEM7 so I may access and download from systems? I have thought about using the ST80III to access the system and downloading in NEWDOS80 then using either TRSCPM.COM or FILETRAN 1.30 to transfer the program to the CP/M 8" disk. Do you think that this will work?

Any assistance rendered will be greatly appreciated. Thank you.

Marvin E. Decker
San Antonio, TX

The best way to get a copy of Modem7 is from a local CP/M user's group. Someone in the group should be able to configure it for your Model I, or a Model I version should be available. Otherwise, check out Kelly Smith's RCPM system (go ahead and use any modem program), and find the MODEMxx.HEX file and download it to your system by viewing it on your screen. Download MODEMxxx.SET also, in order to read documentation on how to patch your MODEMxx.HEX file using DDT.

Thank You

I recently purchased issue #3 of *User's Guide* at a nearby newsstand and was knocked out enough to want to purchase Issues #1 & #2. And also to subscribe for a year beginning with #4.

Thank you very much.

August G. Blume
San Anselmo, CA

I have just had an opportunity to read a copy of your magazine, Vol 1, No. 3 which a friend loaned to me. It is what I have been looking for since I first started to use CP/M (about 9 months ago).

Many of us who are not computer thinkers but users need your sort of information. I have purchased the entire WordStar Series of programs and find great difficulty in reading their manuals. The issue I read cleared up a lot of problems. I have been trying to cope with my modem (Apple-Cat by Novation) and have had nothing but problems. Your article in the same issue has not solved all of my problems yet; but it sure has put me on the right track. All in all, thank you.

Boris A Rapport CLU
Anaheim, CA

An Apology

I have been a subscriber "from the beginning," and am looking forward to the next issue. I do hope, however, that you will be more cautious about your promises for "future issues;" I was counting on your help (via SuperCalc) in preparing my taxes, and that promised article did not appear.

Also, I hope you will be able to dig in more depth into some topics, such as, for example, "customizing" WordStar (preferably, an expansion of the several already-published articles — in *FOGHORN*, *PicoNews*, and *Portable Companion*, that I know of — to really tell us how to adjust the print control commands).

Similarly, I have seen several articles about MODEM7 (in the sources already mentioned), and all the of them fail to discuss the same crucial area, namely how to make MODEM7, or any other modem software, work on the machine which the reader has at home, unless it happens to be either a PMMI or D.C. Hayes Smartmodem. I am finding this much more difficult than following the directions in actually using the program to communicate with a RBBS. I realize that your publication is intended to be a software magazine, but I think that now and then it is necessary to slop over into the area of hardware if you are going to deal adequately with the software.

I expect the next issue will arrive any day now.

W. Duane Austin
Mtn. View, CA

First, an apology. We promised SuperCalc tax models, and we failed to deliver. Several companies reportedly had them available for 1982 returns, but only one has so far arrived, and it was incomplete. We are awaiting models for evaluation at the end of this year.

About Modem7: unfortunately, there are too many machines to describe how to configure Modem7 for any particular one, and too many versions of Modem7 (many of which require different patching instructions). Kelly Smith (CP/M-Net) is purportedly working on the problem of too many Modem7 programs and not enough standard configurations; Byron McKay (of PicoNet) hopes to find time to write up technical details for such an article in a future issue of UG..

Strunk & White

Your intro to CP/M was so clear that I never once had to move my lips. Strunk & White must be required reading in your domain...

Jon Sparks
Future Currents Editor
The Commercial Appeal
Memphis, Tennessee

SuperCalc Summary

A SuperSummary to SuperCalc's Commands, Options and Functions

by Tony Bove & Cheryl Rhodes

In this summary we use *this typeface* to show commands and expressions you type at your keyboard, and this typeface to show displayed messages. In addition, we use these symbols:

↩	RETURN key (CR or ENTER on some keyboards)
↑	Control key (CNTL, CTRL or ALT on some keyboards)

Starting and Leaving SuperCalc

SC ↩

Start SuperCalc. Follow this with another RETURN to reach the blank worksheet, or type a question mark (*?*) for help.

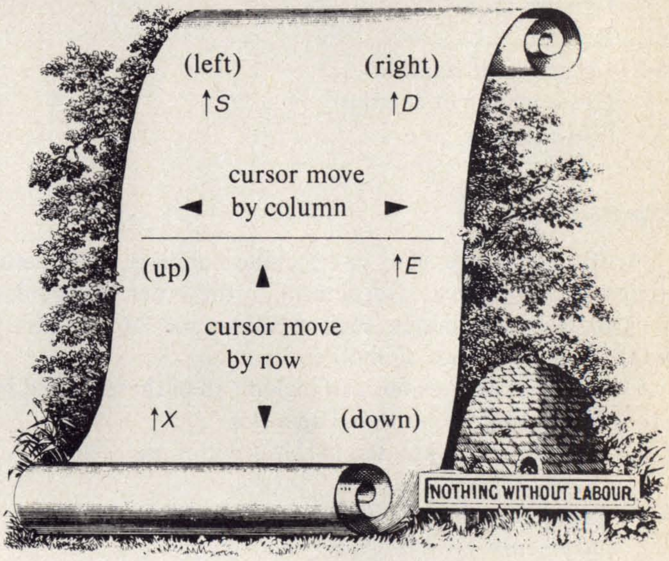
*/Q*uit

Leave SuperCalc. Follow with a *Y* to confirm.

Moving the Cursor

↑*X*	Move down one row (line).
↑*E*	Move up one row (line).
↑*D*	Move to the right one column position.
↑*S*	Move to the left one column position.

Referring to Cells



↩	(RETURN key) In most commands, a RETURN specifies the current cell.
=	Type an equal sign and a cell's coordinates at the beginning of a command line to move directly to the cell.
ESC	(Escape key) In most commands, pressing ESC lets you then use cursor movement keys to move to a cell in order to use its coordinates in a formula or as an answer to a command query. Press ESC again to return to the command line.

Calculations and Formulas

A formula consists of a number, or numbers linked by operators to form an expression. Expressions can be combined with more functions to make a large formula; they can also be combined with the results of another cell's formula.

To force calculation manually, type an exclamation point at the command line (*!*).

Numbers

Any decimal number or integer, with optional plus or minus sign. Numbers can also be expressed in scientific notation (e.g., *1.23E2*).

Operators for Arithmetic

+	Addition
-	Subtraction
*	Multiplication
/	Division
↑	(Caret) Exponentiation

Operators for Conditional Expressions

<	Less than
>	Greater than
=	Equal to
<=	Less than or equal to
=>	Greater than or equal to
<>	Not equal to

Expressions

Arithmetic expressions can include numbers, arithmetic operators, parentheses (to determine the order of calculation) and cell coordinates (such as B23 for row 23, column B) to represent the result of another formula.

Conditional expressions can include all of these, joined in a relationship by a conditional operator.

Expressions are evaluated in this order of preference:

1. Expressions in parentheses
2. Exponents
3. Multiplication and division
4. Addition and subtraction

Functions

All functions require some *argument* to act upon, such as an arithmetic *expression* or the result of another function or formula.

Functions can be used within other functions, as long as you use them within parentheses to properly differentiate their arguments.

In the following descriptions:

value

Result of any arithmetic calculation using expressions, possibly other functions, and possibly the result of another formula.

list

Range of cells in a row or column, such as A1:A20 or B30:G30. A list can also include cells outside the range, as in (A1:A5,A8,A10:A20).

ABS(value)

Produces the absolute value (non-negative number).

AVERAGE(list)

Calculates the average (sum of the list divided by the number of list items).

COUNT(list)

Gives you the number of cells that are not blank and not filled with text (i.e., all cells that have numbers or formulas).

EXP(value)

Raises the number e exponentially, using the *value* as the exponent.

INT(value)

Produces an integer rounded from *value* for display only; *value* is stored as a decimal number.

LOOKUP(value,list)

Finds the next greater value in a list of values that is closest to the *value* you supply.

(LOOKUP actually searches for the last value in the list that is less than or equal to *value*, and returns the next greater value. In a row list, LOOKUP returns the next value to the right of the value it finds; in a column list, LOOKUP returns the next value below the one found.)

LN(value)

Returns the natural log (log base e) of *value*.

LOG10(value)

Returns the log base 10 of *value*.

MAX(list)

Returns the maximum value in a *list* of values.

MIN(list)

Returns the minimum value in a *list* of values.

NPV(discount,list)

Calculates the net present value of a list of cash returns at the given *discount* (*0.10* for 10%). The list should be in positive, ascending order. Net present value is the present value of future cash flow, discounted at the appropriate cost of capital, minus the cost of investment.

PI

Returns the value of Pi to 16 significant digits (3.141592653589793).

SIN(value)
COS(value)
TAN(value)

These calculate trigonometric functions from the radian *value* you provide.

ASIN(value)
ACOS(value)
ATAN(value)

These provide the radian value of the angle described by the trigonometric function.

SQRT(value)

Calculates the square root.

SUM(list)
Returns the sum of the values in the *list*.

Conditional Expressions

If an expression is considered TRUE, SuperCalc places a one in the cell. If it is FALSE, SuperCalc places a zero in the cell. You use the resulting 1 or 0 in your calculations or in other conditional expressions.

In the following, an *expr* is a logical expression involving logical operators linking *values*.

OR(expr1,expr2)
Either *expr1* or *expr2* (or both) can be true to produce a 1; else 0.

AND(expr1,expr2)
Both *expr1* and *expr2* must be true to produce a 1; else, 0.

NOT(expr)
This is true (1) if *expr* is false; else, 0.

IF(expr1,expr2,expr3)
If *expr1* is true then *expr2* is used for the cell; otherwise *expr3* is used.

Creating/Editing a Spreadsheet

"Label
Labels must begin with a double-quote mark (").

/E Edit a cell's label or formula. The cursor keys work on the edit line as follows:

↑X	Delete character at cursor.
↑E	Insert space at cursor.
↑D	Move to the right one character.
↑S	Move to the left one character.

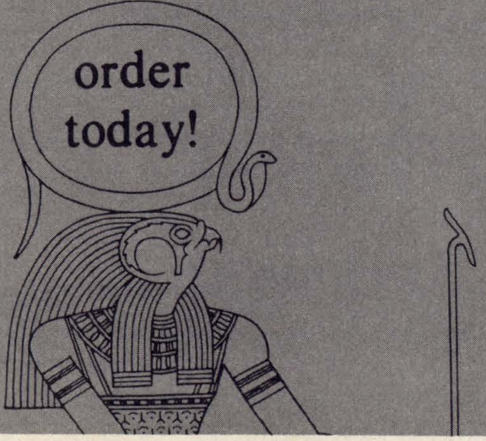
Copying, Moving and Replicating

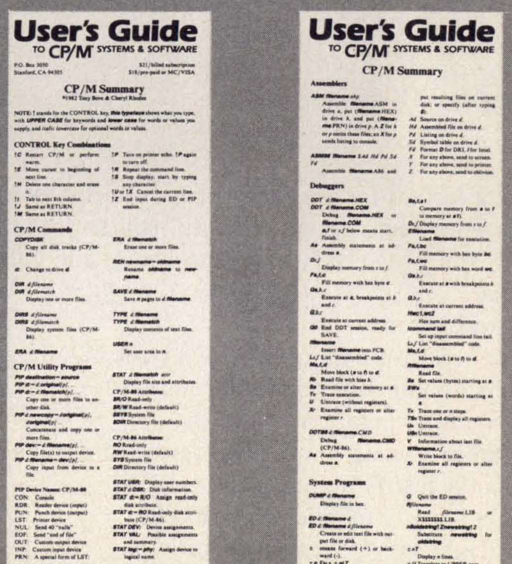
/C opy
Copy one cell, row, column, partial row, partial column or entire block to one other location. Without options, formula coordinates are adjusted for the new location. Options let you adjust specific coordinates and not others, or copy resulting values without formulas:
N o Adjust — Formula coordinates are not changed.
A sk for Adjust — Each coordinate can be adjusted or not adjusted.
V alues — Values (the results of formulas) are copied, not the formulas themselves.

CP/M Summary Card: Only $2

A laminated bookmark-sized card with information on BOTH sides: user's commands and utilities, and programmer's utilities. Covers both CP/M-80 and CP/M-86.

Send a **$2.00** check or money order, or Mastercharge/Visa information, to *User's Guide*, P.O. Box 3050, Stanford, CA 94305.


order today!



Actual size is 12 inches high by 3 inches wide

/*M*ove

Move a column or row to a new location (formulas are automatically adjusted).

/*R*eplicate

Copy one cell, a range of cells, an entire row, an entire column, a partial row, or a partial column to several other locations. Without options, formula coordinates are adjusted for each new location. Options let you adjust specific coordinates and not others, or copy resulting values without formulas (see /*C*opy).

Deleting and Inserting

/*B*lank

Erase contents and formatting for a cell or range of cells, and leave the range blank.

/*D*elete

Delete a row, column or entire file. Move other rows or columns to adjust display and formulas.

/*I*nsert

Insert a blank row or column, and move other rows or columns to accommodate it while adjusting formulas for their new locations.

/*Z*ap

Re-create a blank worksheet (deletes all rows and columns at once). Useful before loading or creating a new spreadsheet.

Formatting for Display

/*F*ormat

Define a format for one cell, a range of cells, an entire row, an entire column, or "globally" (the entire spreadsheet). You can specify a column width, and optional formats:

I — Integer (rounded)

G — Generally used format, which is the best fit: either a decimal number or scientific notation.

E — Scientific notation (using e).

$ — Dollar format: number with two decimal places.

R — Right-justify numbers within cells.

L — Left-justify numbers within cells.

TR — Right-justify text within cells.

TL — Left-justify text within cells.

*** — Display asterisks equal to integer value of cells.

D — Reset cell display to the default for the row or column.

/*G*lobal

Set options for the entire spreadsheet:

*F*orm — Display results of formulas, or the formulas themselves.

*N*ext — Turn off the cursor's auto-advancing feature (usually, every time you press RETURN the cursor moves to the next cell).

B order — Display or turn off row/column border labels.

*T*ab — Allow cursor to move only to unprotected, nonblank cells for data entry.

*R*ow — Calculate across each row first, starting at row 1.

C ol — Calculate down each column first, starting at column A (default).

*M*an — Turn off automatic calculating for data entry, and calculate manually by typing an exclamation point (*!*).

*A*uto — Turn on automatic calculation (default).

Titles, Windows and Protection

/*P*rotect

Protect a cell, partial column, partial row, or block from any deletions, editing changes, copying, replicating, or loading of another spreadsheet.

/*U*nprotect

Remove protection from a cell, partial column, partial row, or block.

/*T*itles

Lock columns and rows as titles in place on screen while scrolling, or clear the titles. You have these options:

*H*oriz — Lock current row and all rows above it.

*V*ert — Lock current column and all columns to the left of it.

B oth — Lock current row and column, and all rows above and all columns to the left.

*C*lear — Remove lock (clear all locked titles).

/*W*indow

Split display into two parts, at the current row and column. Each part can be displayed in different formats. Switch the cursor from one part to the other by typing a semicolon (;).

You have these options:

*H*oriz — Split horizontally at the current row.

*V*ert — Split vertically at the current column.

*C*lear Split — Clear the split screen effect.

S ynch — Synchronize scrolling in both parts of the screen.

U nsynch — Turn off synchronized scrolling; only the window with your cursor moves when scrolling.

Printing Spreadsheets

If you want to print spreadsheets without the row/column borders, use the /*G*lobal command with the /*B*order option to turn off the border display before printing.

*/O*utput

Send all or part of the spreadsheet to the printer, to the console display, or to a disk file. First options are the type of report:
*D*isplay — Spreadsheet as it appears on screen.
*C*ontents — Contents of cells with format and data type, one cell per line.

Next options are:
*P*rinter — Send directly to printer.
*S*etup — Setup printer with more options (below).
*C*onsole — Send directly to console display.
*D*isk — Send directly to disk file with the assumed extension ".PRN". See saving and loading for options on overwriting or backing up files.

Setup options are:
*L*ength — Number of lines per printed page.
*W*idth — Number of characters per line.
*S*etup — Send special setup codes to printer.
*P*rint — Send directly to printer (use after changing other setup parameters).

Loading/Saving Spreadsheets

*/L*oad

Load a spreadsheet from a disk file (assumed to have ".CAL" extension if you don't specify one). Type a filename, or press RETURN to display these options:
*C*hoose — Choose another disk drive.
*D*isk — Display directory of all files.
*S*uperCalc — Display only SuperCalc spreadsheet files (assumed to have ".CAL" extensions).
↑*Z* — Control-*Z* to return to spreadsheet and command line.

Type a filename, and you have these options:
*A*ll — Load the entire spreadsheet.
*P*art — Load part of the spreadsheet (specify range to load from, and location in worksheet to place the loaded section). Options are the same as for */C*opy.

*/S*ave

Save a spreadsheet in a disk file. Type a filename, or press RETURN to see the directory (as shown with */L*oad). Spreadsheets are assumed to have a ".CAL" extension (unless you specify one). After typing a filename, SuperCalc searches for a file of the same name. If there is one, you have these options:
*C*hange name — Change the name of the file you are saving, to avoid overwriting the old file of the same name.
*B*ackup — Change the name of the old file to have a ".BAK" extension, and save the new file under the regular name, without overwriting the old file.
*O*verwrite — Overwrite (destroy) the old data in the file, and save the new spreadsheet with this name.

Automating SuperCalc Operations

Create a text "command" file containing *exactly* the letters you would type on the SuperCalc command line, including options and places where you would press RETURN. If you want to include the */E*dit command, use it only as the last command in this command file.

Name this file with an ".XQT" extension (you can use any extension, but SuperCalc assumes ".XQT" for this type of file).

*/X*ecute

Execute the commands in a command file. Type the filename (SuperCalc assumes the ".XQT" extension unless you specify one) of the command file, or press RETURN to see a disk directory (described with */L*oad).

SC filename ↩

Execute a command file from SuperCalc start-up. Use this method to load a command file containing the entire instructions for setting up a spreadsheet. Also use this method for "batch processing" operations, where *SC filename* commands are used in a CP/M SUBMIT file.

@
can become
HENRI DE TOULOUSE-LAUTREC

IMPOSSIBLE? NOT WITH
SMARTKEY!

SMARTKEY™ is a unique utility that can redefine any ASCII character or function key to become anything you want. For example, "#" becomes "pip b:=a:*.pas[v]". With a single stroke, a key can represent a chosen character or string at the system level or within a program. Instantly. Without rewiring or soldering.
SMARTKEY™ is completely transparent to the user. It resides on the top of memory and intercepts calls to the BIOS, translating system input to whatever you desire. You can even change a key definition while another program, such as WordStar™, is in operation . . . without interruption! It's perfect for programming, data entry or word processing.

"EXCELLENT" InfoWorld
"VERSATILE AND RELIABLE" Lifelines
"WORKS LIKE A CHARM" Microsystems

Think of the acceleration in productivity. Think of the versatility in keyboard layouts. Think of the possibilities. And, best of all, **SMARTKEY™** is only $60.

Ask about **SMARTPRINT™**, **SMARTSCREEN™**, **SPOOL™** and other programs.

To order or obtain more information, call or write to:
HERITAGE SOFTWARE, INC.
2130 S. Vermont Ave., Los Angeles, CA 90007/(213) 737-7252

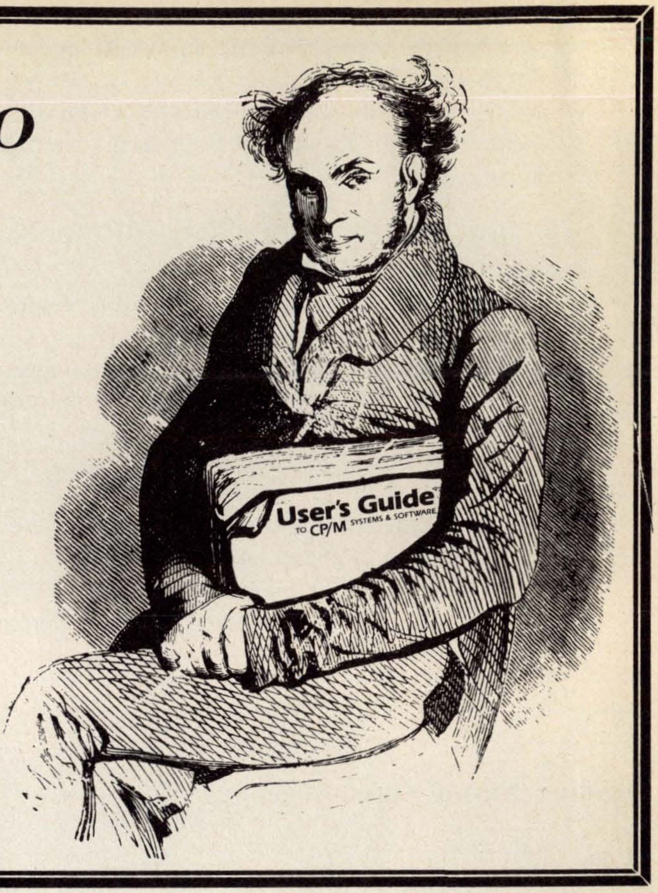
SMARTKEY™ is compatible with all standard versions of CP/M.™
Programs copyrighted by FBN Software.
WordStar™ is a registered trademark of MicroPro, Inc.
CP/M™ is a trademark of Digital Research.

Circle reader service number 8

You will have to buy your own User's Guide.

If you are one of the 700,000+ users of CP/M systems and software, and you don't have a copy of *User's Guide* #1, order back issues today! Our first printing of 10,275 copies means that only one in 70 CP/M users will receive a copy of this collector's item. Use the subscription card in this issue to order at $4 each.

If you are interested in article reprints, write to: *User's Guide* Reprints, P. O. Box 3050, Stanford, CA 94305.



Product Announcements...

This section provides information on new products and a service that provides more information from the product manufacturer. Each item has a reader service number. To receive by mail more information about that item from the manufacturer, simply circle the same number on the postpaid reader service card inserted in this issue, and drop the card in the mail. We give your name and address to the appropriate manufacturer, who sends you the information.

The service helps you get detailed information about products you may want to purchase. We do not sell products ourselves, nor have we evaluated products in this section. We may have reviewed some, but we have not had an opportunity to review all of them, since most of them are new and perhaps unavailable for a few weeks or months. We can't guarantee that none of these hardware or software products will actually be "vaporware" (nonexistent), but most of the products mentioned here are from reputable companies and we are confident in their abilities to produce excellent products.

Computers, systems, software and peripherals

New Computers:

PC-8800 From NEC

NEC Home Electronics introduced the PC-8800, a complete eight-bit CP/M system for $2497, which includes a bundle of software: WordStar and MailMerge (word processing and form letters from mailing lists, from MicroPro), Multiplan (spreadsheet program from Microsoft), and two versions of NEC BASIC (N80 and N88).

PC-8800 systems can be upgraded to sixteen-bit systems with an additional circuit board. 5.25-inch or eight-inch floppy disk drives are available; the unit comes with one 5.12-inch disk drive, a cassette port, a monitor port,

Continued on page 36

CP/M Plus: The Newest Version of Single-User CP/M-80

CP/M Plus For Eight-Bit Systems Looks Like MP/M, Yet Is Compatible With Earlier Versions of CP/M

by Steve Rosenthal

© 1983 Steve Rosenthal

CP/M Plus® is a major update of Digital Research's CP/M® operating system for eight-bit computers. Released late last year, it also carries the name CP/M 3.0.

This article is the first part of an overview of Digital Research's new CP/M Plus. This installment describes the overall system, and how it appears to a casual user. The next installment will continue with a more detailed look at the inner workings of this new operating system. Accompanying this article are two boxes: one on compatibility, and one on documentation. In addition, this issue has a reference guide to CP/M Plus user commands.

Like the eight-bit CP/M versions before it, CP/M Plus is designed to be run on computers using the Intel 8080 and 8085 and the Zilog Z80 chip families. It does require more memory than previous versions of CP/M-80, but it provides greatly expanded features and better performance.

First-time computer users will find CP/M Plus easier to use than previous versions of CP/M, especially if they use off-the-shelf software. Users with floppy disks will appreciate the faster running speed of programs that use many disk operations. Experienced programmers will find more useful tools provided by the operating system. Computer manufacturers and system integrators will like the smooth upward migration from previous versions of CP/M.

Although many of the limitations found in CP/M 1.4 and CP/M 2.2 have been corrected, there are a few things we would have liked to see, but could not be provided without making the system incompatible with previous versions. Documentation is still weaker than it should be (see accompanying box), and there are still a fair number of cryptic error messages.

Typeface Conventions

In this article we use *THIS typeface* to show commands and expressions you type at your keyboard, and this typeface to show displayed messages.

In addition, we use these symbols:

↩	RETURN key (CR or ENTER on some keyboards)
↑	Control key (CNTL, CTRL or ALT on some keyboards)

Banked and Non-Banked Versions

CP/M Plus is primarily intended for eight-bit computer systems with more than 64K of memory. Although eight-bit computers can use a maximum of 64K bytes at any one time, extra memory can be organized into more than one *memory bank*. The computer can be set to turn on each bank as required. This allows the operating system to leave most of one bank entirely for user programs, and to hide most of its operations in another bank. As a result, the operating system can be larger without taking away from your program space. This has allowed Digital Research to add a number of features which would have taken too much room in previous versions of CP/M-80.

The company decided to also market CP/M Plus in unbanked versions, which need at least 64K of memory. Some features are left out of the unbanked system to save space, but the effort to keep the two versions compatible has meant that the banked version does not make as good use of the greater availability of memory as it could have.

Overall Features

The primary purpose of any operating system is to load and store user programs and data, and to give users commands to use to move programs and data to/from other disks and devices (modems, printers, etc.).

CP/M Plus accepts and executes commands entered from the keyboard of your "console" (almost always a video terminal). Some CP/M Plus commands are built-in, and others are programs stored on disk (".COM" files).

The six built-in commands are:

- DIR — show the directory,
- DIRSYS — show system files (not shown by DIR),
- TYPE — type (display) the contents of a file,
- REN — rename a file,
- ERA — erase a file, and
- USER — set the current user area number (0-15).

Included in CP/M Plus is an easier-to-use SAVE utility, implemented as a special program (discussed below).

In addition to these built-in versions, each of these commands (except USER) has a longer, more sophisticated non-resident version. These expanded versions are provided on disk as .COM files, and CP/M Plus automatically picks the on-disk version if you request any options with the command.

There is now a HELP program which provides basic information about the various system programs and options. It can be used from from the command level, but not during the execution of a user or system program.

Data and files are assumed to be stored on floppy or hard disks. CP/M Plus supports up to 16 drives (A-P) with up to 512MB (megabytes) each, and each file can be up to 32 megabytes long (a megabyte is over one million characters). In addition to keeping track of where each file is stored on the disk, CP/M Plus can record time and date information showing when files were created, last accessed or changed.

For faster disk access, CP/M Plus can read or write files in large chunks. If there is enough available free memory, the system buffers disk data in memory, making access to recently-used information almost instantaneous.

Another new feature is the SETDEF utility, which lets you define a program search path. If the desired program is not found on the current drive and user area, the system will look in various other drives and user areas. You can also assign a drive to hold temporary work files.

Output usually displayed at the console can be directed to a file with the PUT utility, and keyboard input can be supplied from a file with the GET utility. Both utilities can work together to do "batch" (unattended) processing and some of the simpler program-to-program data transfers that UNIX-users calls "pipes."

In the banked version of CP/M Plus, disks and files can be password protected. Use the SET utility to specify a password that is needed for reading, writing, or deleting each file.

CP/M Plus allows most types of errors to be handled by system programs (i.e., outside of the system's BDOS module). If computer manufacturers or system integrators make use of this feature, they can make errors easier to handle than in previous versions of CP/M.

The banked version provides command line editing for the errors you make when you type commands. If you type a wrong value at the end of a command line and the system rejects the line, you can use certain control keys as editing keys to change any part of the line.

Unlike previous versions of CP/M, CP/M Plus does not assume the typical user has an ancient teletype hard-copy terminal attached to a computer with only two floppy disk drives, or that several users may try to share one system and may change disks without the knowledge of other users.

When the system notices that a disk has been changed (because the computer senses the change, or because the system knows when it checks the directory), the system no longer protects the disk from writes and updates, assuming that someone has switched a disk while the other users were not looking. Now, the system automatically logs-in disks as found, and only protects them from writes and updates when you set them explicitly to read-only status.

Because the system assumes you are using a video terminal, CP/M Plus can pause after filling the screen with output, giving you a chance to read those vital messages before they vanish. You can turn this paging on and off using the PAGE and NOPAGE options of the TYPE command.

There are changes that some users may not like, such as the changes to the SAVE command, and the expanded utilities that occupy disk space. Because the expanded versions of the built-in commands are now utility programs occupying space on disk, each time you make a new system disk you have to remember to copy the files corresponding to these commands, if you want to use them.

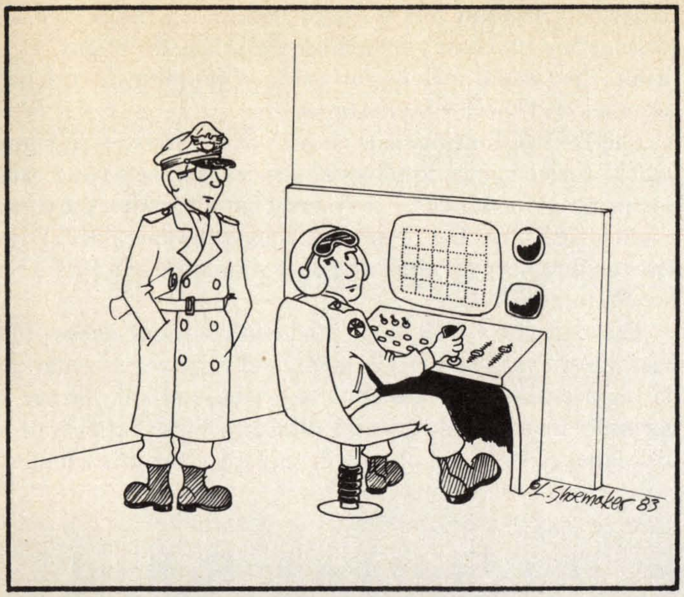
The SAVE command cannot be executed as an afterthought. To SAVE an area of program memory on disk, you must use the SAVE utility first, run the program, and then return to the command level. Since programmers use this utility more than users, the requirement does not affect ordinary users. Programmers may actually consider this new SAVE to be an improvement over CP/M 1.4 and 2.2 versions — no more re-booting, saving memory, and then trying to recover those precious lines or numbers that were left dangling when a program unexpectedly went south.

All CP/M Plus command options start with the left square-bracket (*[*). If your left square-bracket is not in a convenient place on your keyboard, you'll soon realize that almost every system utility has options you will find useful. Maybe you should replace your keyboard, or use a program like SmartKey® (Heritage Software) to re-program the keys.

System Start-Up

Assuming your computer and terminal are both on, insert the system disk in the system drive (usually drive A), press your reset button, and wait for the familiar CP/M sign-on message to appear on your screen.

After it displays the sign-on message, CP/M Plus looks for a special file called "PROFILE.SUB." If it finds such a file on the disk in drive A, CP/M Plus executes the file's commands just as the SUBMIT utility would execute them.



"The color's down, General... We're going to have to fight in black and white!"

(".SUB" files are text files that contain the names of commands or programs to be executed in "batch" mode). CP/M Plus looks for PROFILE.SUB and executes it every time you restart the system.

You can use any text editor to create PROFILE.SUB, or you can use the enhanced SUBMIT utility to create the file. PROFILE.SUB can consist of commands to invoke a standard housekeeping routine, start up a user program, or perform system tests. Although the system lets you select baud rates, PROFILE.SUB is not executed until after the system sign-on, and therefore, you cannot set the initial baud rate for your system console in PROFILE.SUB.

If there is no PROFILE.SUB file, CP/M Plus resumes normal operation.

Command Structure

CP/M Plus uses the standard CP/M command structure. The "user interface" (known as the CCP module or Console Command Processor) tells you when it is ready for a command by presenting the letter corresponding to the current disk drive and a prompt:

```
A⟩
```

You can select one of 16 user areas, which are labels under which disk information can be stored. The prompt shows if any user area other than 0 is current by preceding the disk with the user number. For example, if you had selected user area 5 on disk C, the prompt would be:

```
5C⟩
```

To execute a command, you respond to the prompt with the command name, followed by a space and then any supplementary information such as the names of data files or command options. Although each command defines its own syntax for its "command tail," most commands follow the basic structure. Here's a typical command:

```
B⟩PIP B:=A:*.* [A] ↩
```

This command runs a copying program named "PIP" using the "command tail" `B:=A:*.* [A]` The command tail follows the PIP program's rules: first you specify the destination (in this case, disk drive *B:*), followed by an equal sign and the source (in this case, all files on drive *A:*), followed by an option or parameter (in this case, *[A]* which stands for the Archive option).

Built-In Commands

The DIR command (show the disk directory) is probably the most used well-known utility program. Its simple form is still resident in the operating system for fast response. The DIRSYS (or *DIRS*) utility shows the directory of programs and files marked as "system" files. DIRSYS is a great convenience if you put different systems on different disks. However, there are over a score of options available in the disk-resident version of DIR and DIRSYS, many of which deserve regular use.

Among the options I find most useful are the USER=ALL option (which shows all files on the disk, regardless of whether they are stored in the current user number), the FF and LENGTH options (for listing the directory on my printer), and the EXCLUDE option (which shows everything which *doesn't* match a given file pattern).

The default for the expanded directory command is a listing in alphabetical order, with added information depending on any options you specify. The options replace most of the old STAT command features available in CP/M 1.4 and 2.2.

The TYPE command (used to list the contents of files) has only two options, PAGE or NOPAGE. With paging, the system pauses after a screen is filled with output, and continues when you press any key on the keyboard. If you want to change the screen's "page" length, you must first use the SET utility before you use the TYPE command. (You may need to change the page length for any files you want typed by your printer.) The CP/M Plus version of the TYPE command lets you type the command without expecting you to also specify a filename. The TYPE program will prompt you for the filename if you don't specify one.

ERASE in CP/M Plus has some safeguards to prevent you from accidentally wiping out something you really want to keep. If you want to erase all files matching a certain pattern, and you provide a filename match, ERASE asks you to confirm before erasing any files. I find this feature more annoying than useful, but on the other hand my unERASE program that I relied on to reverse my accidental erasures does not work on CP/M Plus.

You can also give the option CONFIRM, and ERASE then asks you to confirm before erasing *each* file. This option is a useful one for beginners and for those who do not have programs that can recover accidentally-erased files.

RENAME will change the name of one file, or a group of files. If the new name is the same as an existing file, you are given a choice of erasing the existing file or not using the new name for the file you are trying to rename. If you type the RENAME command by itself, you will be prompted for the needed filenames.

The USER command can also be given just as a single word, and you will be prompted for the required entry. While some CP/M users in the past have found user areas to be more trouble than they were worth, improvements to CP/M Plus should change that. User areas do come in handy when you have a large hard disk.

Setting System Parameters

Design by evolution has left CP/M Plus with a large, somewhat confusing mix of system parameters, flags, console (display and keyboard) characteristics and user options. As a result, the system includes an array of programs to set parameters and check system status.

The DEVICE utility sets console characteristics, assigns logical device names to physical devices, and sets baud rate and protocol for devices. For system characteristics, the command is SETDEF. DATE sets or displays the date, INITDIR sets the directory for time and date stamping, and SET sets file characteristics.

The expanded DIR command with options shows file characteristics. The SHOW command displays information on the overall status of disks and drives, and lets you see a normally invisible entry in the disk directory that acts as a disk label (CP/M 2.2 users will recognize the other half of

The Documentation Is Better, But Still Lags Behind

The good news is that CP/M Plus documentation is a tremendous improvement over prior versions of CP/M-80. The bad news is that it still has a long way to go.

The basic documentation is now divided into four manuals (a User's Guide, Programmer's Guide, Programmer's Utilities Guide and a System Guide). Although some material on SID (Digital Research's Symbolic Instruction Debugger) is also included in the User's Guide, my package came with a SID manual as well.

Three of the system manuals came as loose-leaf sets, along with a fancy three-ring binder with matching box. The System Guide is the same format as the other manuals, with holes punched for rings, but is bound (and there is no room for it in the binder).

All the manuals have indexes, and there is even a glossary in the User's Guide. Unfortunately, in the User's Guide the page numbers in the index are often off a page or two.

Overall, the organization of the manuals seems logical, and the writing is generally clear. The layout is straightforward and the design is clean but not exciting. The text was apparently produced on a daisy-wheel printer without a proportionally spaced printwheel, so the type is legible but not crisp. Subheads and examples are both in daisy-wheel bold (probably double or triple strike), which makes it hard to tell them apart in a hurry.

The System Guide is not an overall guide to the system, but a guide to constructing and installing CP/M Plus on a new machine. It contains an overview of the system requirements and memory organization. Among the topics covered are what must be included in a user BIOS, how to bring up a system starting from CP/M 2.2, and how to link together the user portions with the standard parts of CP/M Plus. An appendix includes a sample of the needed BIOS sections for a Z80-based machine.

The User's Guide is the main tutorial and reference manual on how to run the system and the housekeeping utilities. If you are already familiar with CP/M, the explanations will be clear. If not, even though the style and content is vastly improved you may still need some help. While the commands are clearly explained and each explanation is accompanied by examples, there is no clue as to what you first should know to start using the system.

The Guide starts with a general introduction, and then instructions for backing up the system disk. It continues with a more complete explanation of files and file naming, followed by a section on the console and printer.

The command summary section includes both the general rules for commands, and a detailed explanation of most of the system utilities. ED, the CP/M editor designed for a hard-copy teletype, is still hanging in there, and rates a chapter all its own. The appendices cover a range of useful information, including a complete list of the several dozen possible error messages, and what could be responsible for their appearance.

The Programmer's Utilities Manual is really a manual for the MAC macro assembler and its variations and supporting programs. The information is not specially tailored for CP/M Plus, but anyone dedicated enough to learn macro programming and linking should be able to place all the instructions in the correct context.

The Programmer's Guide is the key to the system calls and how programs run under CP/M Plus. There is a good general introduction, but the detailed discussions of the calls are somewhat obscure at many points. There are many calls with a lot of details, and the manual is definitely not bed-time reading.

The manual editors did go to the trouble of reproducing several of the vital tables more than once, so you can refer to them in conjunction with applicable system calls. Unfortunately, each version tends to be slightly different, and some leave out important information found in others.

Hidden away in the back of the manual is a short section on construction RSX (resident system extension) files. These files allow the customization of the operating system, and allow some of the system programs to perform special functions (More about RSX's in the next installment).

STAT in the SHOW program). DATE displays the date.

People using CP/M as a framework for an application package will not have to bother with these utilities. The rest of us will have to buy the inevitable reference cards, memorize a lot of possibilities, keep *User's Guide* within reach, and work on our own HELP messages with the HELP command (see below).

Although these utilities have error messages to tell you if some essential information is missing or the syntax is incorrect, no lists of possible options are provided within the programs. You can memorize all the several dozen options, look at the manual or the on-line HELP program, and then call the right program and remember the exact option.

Until you memorize the more common options, a menu-driven program (one that presents the possibilities and lets you choose between them) would be of help. I would be amazed if there aren't a dozen programmers writing such programs, either for sale or for some user group.

The HELP Program

The HELP utility is Digital Research's acknowledgement that CP/M Plus is too complex for most of us to memorize immediately. Although the HELP utility is meant to serve as an on-line reference, it is more like an on-screen reference card; however, it cannot be used from within other programs.

The HELP system actually consists of two files: a HELP.COM command file (the program), and a HELP.HLP data file. The HELP.COM file presents the material in the HELP.HLP file, and lets you modify or add to the material presented.

In operation, HELP is a modified menu tree providing a list of topics, and it often gives both text about the topic and a list of subtopics. Subtopics can have sub-subtopics, and so on.

For example, you could first type:

```
A>HELP↵
```

You would get the main list of HELP topics and the `HELP>` prompt message. If you want more information about the SET parameter program you would type:

```
HELP>SET↵
```

The HELP system shows you a screen-full of information

Compatibility With Previous Versions, and With More Advanced Versions

Although most programs that run under prior versions of CP/M-80 will run unchanged on CP/M Plus, there are some important exceptions.

The most important is that with banked systems, programs that use direct calls to the BIOS module of the system will rarely work. Although previous versions warned against making such direct calls, even some of Digital Research's programs used such calls. The appropriate method is to use calls to the BDOS module, not to the BIOS module.

The problem is that at least part of the BIOS is probably in the system bank, and the user program is probably in another bank. Information on which parts of the BIOS is in which bank is not available to user programs.

The programs that are most often affected by this change are ones that attempted to read and write to the disk without going through the BDOS. This includes many disk copy programs, disk viewers, catalog programs, command shells ("menu" programs) and programs for generating files for non-CP/M systems.

A more subtle compatibility problem is the expansion of the reserved character list. These are the characters which are taken by the CCP or BDOS parts of CP/M to signal special conditions or the start or ends of fields. As a result, they cannot be used as normal characters in filenames (or passwords).

The catch is that many of us used characters in our filenames that were not formerly reserved. Now, we find that specifying these files in CP/M Plus causes strange results.

For instance, many of my files have names including the exclamation point ("!"). However, in CP/M Plus, the exclamation point is used to separate multiple commands on the same physical line. Many of the system programs behave strangely when I use these filenames.

Compatibility With Advanced Versions

Although users of CP/M 1.4 or 2.2 should have no trouble adapting to CP/M Plus, users of more recent Digital Research operating systems will have even less trouble. Digital Research has taken many of the added features developed for their more recent operating systems and brought them back to the single-user eight-bit world.

File protection and time and date stamping were introduced with MP/M-80, and the password protection scheme is compatible with MPM II. Similarly, making the CCP module a transient file comes from MP/M.

Many of the housekeeping utilities are also the products of this "backwards evolution." Only the simplest commands, such as DUMP and USER, seem to have carried forward unchanged. The others are entirely new to single-user CP/M, or have had significant additions.

about the SET program. At the end of the material, you are asked to pick a subtopic from a list of about half a dozen available subtopics.

In a case of what can only be described as "user-unfriendly" programming, you have to precede the subtopic name with a period to indicate to the system that you are answering with a subtopic. HELP then shows you the subtopic material, and so on.

Unfortunately the only way to traverse the tree is from root to branch to leaf. If you read one subtopic or sub-subtopic, you can't go back one level and pick another — the only exit is back to the main HELP prompt and the main topics.

The good news about the HELP system is that it is extensible — two options on the HELP command let you change the HELP.HLP file back and forth into an ordinary text file, and you can add to or change the material as you like.

Programming Utilities

The program development utilities supplied with CP/M have also been upgraded from CP/M 2.2 to CP/M Plus. Included in the standard package are the MAC macro assembler, the RMAC version of MAC for producing relocatable code, LINK (for combining the REL files produced by MAC), and the SID debug and trace program using program labels.

MAC and SID are upgrades of the previous ASM assembler and DDT debugger, and are significant improvements. When an assembly-language program is assembled with MAC, a symbol file is produced along with the optional ".HEX" and ".PRN" files. SID can use this symbol file to show program labels and symbol values during program trace, and accepts symbolic names for parameters.

So with MAC and SID, you can assemble a program and debug it using names such as INIT and PDEC instead of writing little notes such as "the initialization is at address 5C21H and the print decimal routine starts at 42FEH."

Although most new system designs use the Z80 instruction set rather than the 8080 instruction set, Digital Research does not have a decent Z80 assembler. Instead, the company supplies a set of macros to enable MAC to understand a form of the Intel 8080 opcodes extended to the Z80 (for compatibility with the 8085 processor used in many systems, as well as for compatibility with older systems using the 8080). The corresponding debugger, ZSID, is not supplied with CP/M Plus.

Apparently, ".HEX" files will be with us always, even though their original purpose was to allow the Teletype paper punch to act as intermediate storage (".HEX" files store each character of the machine code output of the assembler as a pair of hexadecimal characters).

MAC translates assembly language to a ".HEX" file, and HEXCOM is used to turn ".HEX" files into ".COM" files that can be executed like commands. I've patched my assemblers to go straight from assembly language to produce a ".COM" file, unless you request a ".HEX" file. It would be nice if MAC could be updated to include this feature.

The Search Chain

One of the nicest new features of CP/M Plus is the drive search chain. This lets you say that if a requested program or ".SUB" file is not in the current user area of the current disk, the system should look in a few other places before reporting that the file cannot be found.

The default search chain is to look first in the current user area and drive. If the program is not found, the system looks in user area 0 of that same drive for a copy of that program with the "system" flag set in the program name. So programs marked as "system" files in area 0 are now available to all user areas on a disk.

Using the SETDEF command, we can change the drive chain to start with any disk, and continue searching for up to four disks. All four can be specified absolutely, or one can be set as the current default disk. For each disk, the search is made first in the current user area, and then in user 0 for a file marked "system."

For floppy disk systems, a multi-drive search chain with anything but current drive and disk A should be used with certain reservations in mind. On most implementations, the system "hangs" if one of the drives in the chain is not ready or has no disk in it. On hard disk systems, especially those where one physical drive is defined as multiple logical drives, the search chain can be a real convenience.

Passwords

In banked versions, disks and files can now be given passwords to prevent casual misuse. The system is not designed to discourage serious attempts to thwart protection, but to discourage snooping by unauthorized business employees, family members or fellow students. Passwords generally go after the file name in commands:

```
3A〉MYPROG FILE.DTA;PASSWORD↵
```

This command executes MYPROG.COM using FILE.DTA for data. *PASSWORD* is also supplied in let MYPROG read FILE.DTA.

Passwords are stored on the disk as extra directory entries for each file. Using passwords will reduce slightly the number of possible files that can stored on each disk. The entries are scrambled, so the password cannot be retrieved by simply looking at the entry with a disk viewer program.

Using the SET program, you can also declare a default password. The default password is used by the system whenever one is required and you have not provided one with the filename. The default password is useful for setting up someone else to access your files without divulging actual passwords.

Incidentally, to use password protection, the disk itself must be given a label with the SET program, and the label given a password. If you know the disk password you can change it or erase it; however, if you do not know your disk password, you are locked out of the disk.

Conclusion

CP/M Plus is a step in the right direction for eight-bit single-user systems. Most eight-bit systems are evolving to have more than 64K memory using memory banks, and CP/M Plus provides the capabilities that were previously available only through modification of earlier versions of CP/M.

CP/M Plus also provides features for hard disk users that were previously unavailable, including support for hard disks, larger files (up to 32 megabytes per file), password protection (banked version only) and options for directory displays. Finally, some of the most useful features of an operating system — the ability to provide keyboard input from a file, for unattended operations, and the ability to send output to a file rather than the printer — are welcome additions to CP/M and make CP/M a more productive single-user environment. ⧉

Corrections

The most glaring error is of omission. Many of you were expecting articles on CBASIC, WordStar, CP+ and other utility programs as well as an FMS-80 tutorial.

We had to pre-empt these articles until the next issue to make room for part one of our CP/M Plus reference guide. CP/M Plus is becoming available on popular machines rather quickly, and it has features that have never been explained to our readers; therefore, we felt our readers would be better served with timely reference information.

Also, we quite simply ran out of space. The FMS-80 tutorial, WordStar part two, and CBASIC article are much larger than we thought; the CP+ article seems to have momentarily disappeared into the twilight zone of Kelly's computer, and the rest... well, we are rather ambitious in thinking we'd have room in #4 for all of this. We're learning, and thanks for having patience with us.

Our apologies to Kim Levitt and Frank Wancho for misspelling their names in "CP/M'83 Report" in issue #3. Also, we erred in presenting a "scroll" illustration on page 19 of issue #2; we made a mistake and put in the same "scroll" used for a WordStar tutorial in the same issue. We present the correct version in this issue, accompanying the "SuperCalc SuperSummary."

$29.95
full-feature text editor

OMNIWRITER™

The Ideal Tool for Writers, Editors, Typesetters, and Programmers

Multiple Print Options allow you to print any portion of your file. OMNIWRITER also permits you to edit one file while printing another.

Search and Replace allows you to find any character sequence (including "wild cards"). Make as many as 100 different changes in one pass using **Multiple Global Search.**

Multiple File Access allows you to edit up to three files at a time. Select names from a list and merge them with a letter in another file, or move subroutines from one program to another.

Communications: is both batch with XON/XOFF protocol, and interactive for timesharing.

User-Programmable: OMNIWRITER's buffer files store copy and/or sets of commands. Create a library of text-processing programs to fit your own needs. Commands include **Conditional Test and Branch,** which makes complex reformatting simple. Perform **code conversions** from 8080 to 8086 assembly language.

Safety Features: Interrupt any automatic process by simply hitting any key on the keyboard. Also, OMNIWRITER holds all deleted copy in a temporary buffer, from which you can retreive it if needed.

100-page Teaching Manual, 32-page User's Guide, Installation Program included.

☐ 8″ SSSD for CP/M-80 (ver. 1.4, 2.2), MP/M
☐ 5¼″ disk for CP/M-86 (IBM PC and others)
☐ 5¼″ disk for MS-DOS (IBM PC and TI PC)
Other formats available. Inquire.

Send $29.95 plus $3 postage (add $1.95 sales tax in CA) to:
Omni Systems International, 2229 McGee Avenue,
Suite H, Berkeley, CA 94703. (415) 644-1896.

Circle reader service number 10

TALK WITH THE STARS
COMMX
HP125 1000
VAX
RBBS
PDP 11
UNIVAC
TELEVIDEO
COMPU-SERVE
DEC 10 20 VT18X
TRS 80
THE SOURCE
GODBOUT
MORROW
KAYPRO
ZENITH
XEROX
ALTOS
TELEX TWX
APPLE Z80
ITHACA
NORTH STAR
OTRONA
IBM TSO
MONROE
IMS
OSBORNE
CROMEMCO
VECTOR GRAPHICS
SANYO

the best VALUE anywhere
COMMUNICATIONS SOFTWARE PACKAGE
COMMX-PAC™

- COMMX Smart Terminal and File Transfer Auto Dial and Logon Files for Easy Linkup Protocols for Timeshare, MODEM 7, CRC16 FORTRAN version available for Mainframes
- Unattended "Midnite" Electronic Mail Subsystem
- CONSOLX Remote Operation of Your Computer
- Bulletin Board System with Database Access
- Database File Management System Included
- Utilities for Data Encription, Compression, and More
- CPU License $150   Detailed Manual $25

Contact Your Local Dealer or
Call or Write For Free Brochure

HAWKEYE GRAFIX

23914 Mobile, Canoga Park, Ca 91307 • 213/348-7909 U.S.A

Circle reader service number 7

Continued from page 28

and four expansion slots.

Circle number 52 on the reader service card.

Operating Systems, Tools & Languages:

Digital Research (DRI)

Digital Research announced its new low price for the IBM Personal Computer (PC) version of CP/M-86 ($60), which includes the GSX® graphics extension. This extension to the system provides color graphics control for application programs; the controls are compatible with a recognized European standard as well as with AT&T's videotex standard.

DRI also announced DR Logo, an educational language for the IBM PC that incorporates the best of microcomputer Logo implementations, including Apple Logo and Texas Instruments Logo.

Digital Research's Concurrent CP/M-86® is now available for the IBM PC XT; price, $350 for Concurrent, $60 for regular CP/M-86. Both versions include hard disk support (partitioning), direct booting from the hard disk, and the GSX graphics extension.

Bundled with Concurrent CP/M-86 are "packs" of application software for the IBM PC and PC XT systems. DRI, in conjunction with Software Distributors (reported as the largest CP/M software distributor) jointly produced the bundles to include PC Accounting Pack®, PC CBASIC® Pack, and PC Pascal/MT+® Pack. The accounting pack includes the MBA Accounting Package, with general ledger, accounts receivable and accounts payable. The CBASIC pack includes the CBASIC compiler, pro-

Continued on page 52

CP/M Plus™® Operating Tools

Commands and Utilities For Both Banked and Non-Banked CP/M Plus (3.0) Systems

by Kelly Smith
with Tony Bove & Cheryl Rhodes

CP/M and CP/M Plus are trademarks of Digital Research, Pacific Grove, CA.

CP/M Plus (also called CP/M® 3.0) is produced by Digital Research, Inc. (the producers of the original CP/M), and available bundled with many new microcomputers as well as from various stores and distributors. This reference guide summarizes the commands and utilities of CP/M Plus that are used to operate the system, run application programs, and manage files. A tutorial on these utilities, and reference guides on programming utilities, will appear in a future issue.

Symbols and Typeface Conventions

In this reference guide we use *this typeface* to show commands and expressions you type at your keyboard, and this typeface to show displayed messages.

In addition, we use these symbols:

↩	RETURN key (CR or ENTER on some keyboards)
↑	Control key (CNTL, CTRL or ALT on some keyboards)

Control Key Commands

We use ↑ to stand for the Control key (CTRL, CNTL or ALT on some keyboards).

Moving the Cursor

↑A	Move the cursor one character to the left. Banked system only.
↑B	Move the cursor from beginning to end of command line and back without affecting command. Banked system only.
↑F	Move the cursor one character to the right. Banked system only.

Editing the Command Line

↑G	Delete one character at current cursor position if in the middle of a line. Banked system only.
↑H	Delete one character to the left of cursor.
↑K	Delete character at cursor and all characters to the right of the cursor.
↑R	Retype the characters to the left of the cursor on a new line, and update the command line buffer. Use this to repeat the command line without rubbed-out (deleted) characters.
↑U	Cancel the command line and move the cursor to the next line, leaving a # symbol on the command line. You can then use *↑W* to recall the partial command line (up to the # symbol).
↑W	Recall the previous command line if current line is empty; otherwise move the cursor to the end of the line. *↑J*, *↑M*, *↑R*, *↑U* and RETURN update the command line for recall with *↑W*. Banked system only.
↑X	Delete all characters to the left of the cursor.

System Functions

↑C	Stop executing a program when entered at the system prompt or after *↑S*, and show the system how to write to a new or replaced disk. Use this command after changing a disk in any drive, unless you are instructed to change disks by a program.
↑E	Force a physical carriage return without sending the entire command line to the system.
↑I	Same as the TAB key.

↑J	(Line Feed). Move cursor to the left of the command line and send the command to the system. Line Feed in programs usually has the same effect as RETURN (or ENTER), although in communications and in some programs it moves the cursor vertically one row.
↑M	Same as RETURN (or ENTER, CR, Carriage Return).
↑P	Send your display to the printer (echo console output to the list device).
↑Q	Restart screen scrolling after a *↑S*.
↑S	Stop screen scrolling.

Filenames and Specifications

CP/M Plus (banked version) lets you identify files by name and by password. Unlike CP/M 2.2, you cannot use symbols in CP/M Plus file specifications, because CP/M Plus has a special use for them. CP/M Plus identifies each file by its unique file specification (called *filespec* in this reference), which can consist of four parts:

d: Optional disk drive, a single letter from A to P, followed by a colon.

filename
1-8 letters and/or numbers to uniquely identify the file.

.ext
Optional extension to the filename, of a period and 0-3 letters and/or numbers designating a file type.

;password
Optional password of 0-8 letters and/or numbers preceded by a semicolon.

We use *filespec* in the following command and program descriptions to stand for a file specification. A *filename* stands for a filename and extension only; a *filematch* stands for a filename match and extension match only; and a *d:* stands for a drive specification only. The *filespec* argument can include all of these.

The following symbols can't be used in file specifications: ! $ & * ( ) - + _ \ [ ] : ; ⟨ ⟩ , . ? /.

Valid combinations that can used for a *filespec* are:

filename — SAMPLE
d:filename — B:SAMPLE
filename.ext — SAMPLE.TXT
d:filename.ext — B:SAMPLE.TXT
filename.ext;password — SAMPLE.TXT;WHYADUCK
d:filename.ext;password — B:SAMPLE.TXT;WHYADUCK

If you do not include a disk drive letter and colon, CP/M automatically searches the current drive for the file.

Copying The System

COPYSYS↩

COPYSYS copies the system from a system disk to another disk. The new disk must have the same format as the original system disk.

Time and Date

CP/M Plus keeps track of the time while your system is running. When you start CP/M Plus, the time and date is the creation date of CP/M Plus; you must set the real time and date. You must also INITDIR the directories of disks in order to use the time and date as a stamp to identify versions of files.

DATE↩
DATE C↩
DATE mm/dd/yy hh:mm:ss↩
DATE SET↩

The DATE command lets you display and set the date and time of day. Here are some examples:

A⟩*DATE↩*
— displays the current date and time.
A⟩*DATE C↩*
— displays the date and time continuously.
A⟩*DATE 08/14/82 10:30:0↩*
— sets the date and time.
A⟩*DATE SET↩*
— displays prompts for date and time entries.

INITDIR↩
INITDIR d:↩

Initializes a disk directory to allow date and time stamping of files on that disk. INITDIR can also recover time/date directory space.

Device Assignments and Information

DEVICE↩

Display a summary of device characteristics and the logical device names assigned to the physical devices in the system. Example:

A⟩*DEVICE↩*

DEVICE NAMES↩

List the physical device characteristics.

DEVICE VALUES↩

Display the current logical device assignments.

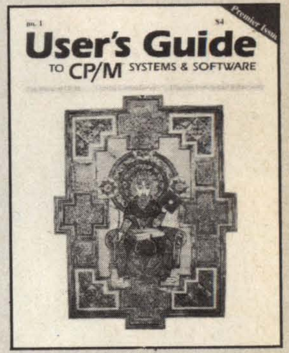
DEVICE device-name ↩

Display the attributes of a particular physical or logical device. Examples:

A⟩*DEVICE CRT↩*
— displays the attributes of the physical device CRT (Cathode Ray Tube).
A⟩*DEVICE CON↩*
— displays the assignment of the logical device CON: (console).

"At last, a magazine with practical, hands-on information for computer users."

— letter to the editors


no. 1
$4
Premier Issue
User's Guide
TO CP/M SYSTEMS & SOFTWARE

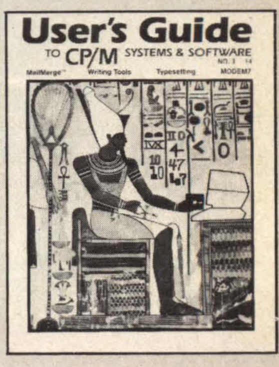
User's Guide™ *is for those who don't care about bits and bytes, but need information on how to make their computers more useful.*

User's Guide *introduces you to the world of CP/M®, the most popular type of small computer system.* **User's Guide** *provides in-depth coverage of CP/M systems and software.*

For example:

- *How to get more mileage out of popular CP/M programs like WordStar®, SuperCalc®, and dBASE II®.*
- *How to gain access to thousands of* **free** *programs in the public domain using a program called MODEM7.*
- *How to recover data from a damaged disk.*
- *How to set up and use a small business accounting system.*

User's Guide
TO CP/M SYSTEMS & SOFTWARE
NO. 2 $4
•WordStar
•Disk Doctor
•SuperCalc


User's Guide
TO CP/M SYSTEMS & SOFTWARE
NO. 3 $4
MailMerge
Writing Tools
Typesetting
MODEM7

User's Guide *features well written articles and clear, step-by-step instructions that help you get the most out of your computer investment.*

For a one year subscription (six issues), use the card next to this ad, or send a check for $18.00 to **User's Guide**, *P.O. Box 3050, Stanford, CA 94305. Foreign subscriptions: $24 pre-paid.*

User's Guide
TO CP/M SYSTEMS & SOFTWARE

CP/M is a trademark of Digital Research.
WordStar is a trademark of MicroPro International.
SuperCalc is a trademark of SORCIM Corp.
dBASE II is a trademark of Ashton-Tate.
User's Guide is a trademark of TUG, Inc.

DEVICE logical=physical↵
DEVICE logical=physical [option list] ↵
DEVICE logical=physical [option list],physical [option list] ... ↵

DEVICE logical=NULL ↵

Assign logical device names to one or more specific physical devices, optionally specifying device characteristics. The option list is one or more options separated by commas and enclosed by brackets. The options are:

XON — The XON/XOFF communications protocol
NOXON — No protocol
baud — Baud rate (transmission speed in bits per second). Acceptable baud rates are 50, 75. 110, 134, 150, 300, 600, 1200, 1800, 2400, 3600, 4800, 7200, 9600 and 19200.

You can disconnect a logical-to-physical assignment using *NULL* as the physical device. Some examples of typing DEVICE commands:

A>*DEVICE CONOUT:=LPT,CRT*↵
— assigns logical console output (CONOUT:) to both the LPT line printer and the CRT screen.
A>*DEVICE AUXIN:=CRT [XON,9600]* ↵
— assigns the logical auxiliary input device (AUXIN:) to the physical device CRT using the XON/XOFF protocol with a transmission rate of 9600 baud.
A>*DEVICE LST:=NULL*↵
— disconnects the logical list output device (LST:).

DEVICE physical [option list] ↵

Set device characteristics for a physical device. Example:
A>*DEVICE LPT [XON,9600]* ↵
— sets the XON/XOFF protocol for the line printer (LPT) and sets the transmission speed at 9600 baud.

DEVICE CONSOLE [PAGE] ↵

Display the current console page width in columns and page length in lines.

DEVICE CONSOLE [COLUMNS=40 LINES=16] ↵

Set the screen size to 40 columns and 16 lines.

Directory Display

CP/M Plus provides enhanced DIR and DIRS utilities (partly in response to the crop of different directory display utilities in the public domain; however, this one performs faster than most of them). Nearly everything you'd expect to find in a directory display is possible with these enhanced versions.

DIR↵
DIRS↵

The DIR command displays the names of all files in the current user area with the DIR attribute, and the characteristics of those files. All files have the DIR attribute by default (which means the filenames are displayed in DIR commands). You can change any file to have the SYS attribute with the SET command (which means the file's name and information are not displayed in a DIR command). The DIRS command (abbreviation of *DIRSYS*) displays all SYS filenames in user area 0 and their characteristics. DIR and DIRS are built-in commands.

DIR d:↵
DIR filespec↵
DIR filematch↵

DIR (and DIRS, if you substitute *DIRS* for *DIR*) will display all filenames on another disk drive if you type a drive that exists (*d:*), or display specific filenames if you type the right name or match. A *filematch* uses * to match any number of characters, and *?* to match any one character to find several files at once. You can precede *filematch* with a *d:* drive specifier. Examples:

A>*DIR B:*↵
— displays all filenames with the DIR attribute in user area 0 on drive B. Note that you could substitute *B:*.** for *B:* in the above example and get the same result.
2A>*DIR C:JANE.TXT*↵
— displays the filename JANE.TXT with the file's characteristics, if JANE.TXT can be found in user area 2 on drive C.
3B>*DIR CHAP?.**↵
— displays all DIR attribute files in user area 3 on drive B whose filename consists of the letters CHAP and one other number or letter (or symbol), followed by any extension.
A>*DIRS *.COM*↵
— displays all SYS attribute files with the extension ".COM" that reside in user area 0 on drive A.

DIR [option list] ↵
DIR d: [option list] ↵
DIR filespec filespec [option list] ↵
DIR filematch filematch [option list] ↵

The DIR command with options invokes a utility program that is an enhanced version of the built-in DIR. This program lets you specify several filenames on the command line, along with one *option list* of one or more options separated by commas and enclosed within brackets. You can usually abbreviate an option to its first three letters:

ATT — displays the file attributes.
DATE — displays date and time stamps of files.
DIR — displays only filenames that have the DIR attribute.
DRIVE=ALL — displays filenames on all on-line drives.
DRIVE=(A,B,C,...,P) — displays filenames on the drives

specified.

DRIVE=d — displays filenames on the drive specified by d.

EXCLUDE — displays filenames that do *not* match the filenames or filename matches specified in the command line.

FF — sends an initial Form Feed code to the printer (if the printer has been activated by ↑*P*).

FULL — shows the name, size, number of 128-byte records, and attributes of the files. If there is a directory label on the drive, *DIR* with *FULL* shows the password protection mode and the time stamps. If there is no directory label, it displays two filename entries on a line, omitting the password and time stamp columns. The display is alphabetically sorted. (See *SET* for a description of file attributes, directory labels, passwords and protection modes.)

LENGTH=n — displays *n* lines of printer output before inserting a heading. Use a number between 5 and 65536 for *n*, default is one screen.

MESSAGE — displays the names of drives and user numbers searched.

NOSORT — displays filenames in the order found on the disk, not sorted in alphabetical order.

RO — displays only the filenames that have the RO (read-only) attribute.

RW — displays only the filenames that have the RW (read-write) attribute.

SIZE — displays the filename and size in kilobytes (1024 bytes).

SYS — displays only the filenames that have the SYS. attribute.

USER=ALL — displays all filenames in all user areas for the default or specified drive.

USER=n — displays the filenames in user area *n*.

USER=(0,1,...,15) — displays filenames in the user areas specified.

Here are examples:

```
A>DIR B: [FULL] ↵
```

— displays full set of characteristics for all files in user area 0 on drive B. Press any key to continue displays more than one screen long.

```
A>DIR B: [DATE] ↵
```

— lists the filenames on drive B and their dates.

```
A>DIR B: [RW,SYS] ↵
```

— displays all filenames in user area 0 on drive B with RW (read-write) and SYS (system) attributes.

```
3A>DIR [USER=ALL,DRIVE=ALL] ↵
```

— lists all files in all user numbers (0-15) on all on-line drives.

```
6B>DIR [EXCLUDE] *.DAT↵
```

— lists all the files in user area 6 on drive B that do *not* have the ".DAT" extension.

```
3B>DIR [SIZE] *.PLI *.COM *.ASM↵
```

— displays the names and sizes of all files with the ".PLI", ".COM" and ".ASM" extensions, that reside in user area 3 on drive B.

```
A>DIR [DRIVE=ALL,USER=ALL] MYFILE.TXT↵
```

— displays the filename MYFILE.TXT if it exists on any active drive in any user number.

```
A>DIR [SIZE,RW] B:↵
```

— displays each RW (read-write) filename with its size in k (kilobytes) found on drive B.

Erasing Files

The improved ERA (or ERASE) command will ask for confirmation if you erase using a filename match, and will also ask for confirmation for each file if you use the CONFIRM option.

```
ERASE filespec↵
ERASE filematch↵
ERASE filespec [CONFIRM] ↵
ERASE filematch [CONFIRM] ↵
```

Removes one or more files from the directory of a disk. You can use the filename match characters * and ? to match several filenames. With the *[CONFIRM]* option, ERASE displays a confirmation message for each filename (*[CONFIRM]* can be abbreviated to *[C]*). When files are erased, directory and data space is automatically reclaimed for later use by another file. You can abbreviate ERASE to ERA. Examples:

```
A>ERASE TONY.TXT↵
```

— removes the file TONY.TXT from the disk in drive A.

```
A>ERA *.TXT↵
Confirm (Y/N)?Y
```

— removes all files with the extension ".TXT" from the disk in drive A.

```
B>ERA A:MY*.* [CONFIRM] ↵
```

— displays each filename on drive A that begins with MY, and asks for confirmation of the erase. Type *Y* to erase the file displayed, *N* to keep the file on disk.

```
A>ERA B:*.*↵
Confirm (Y/N)?Y
```

Removes all files from drive B.

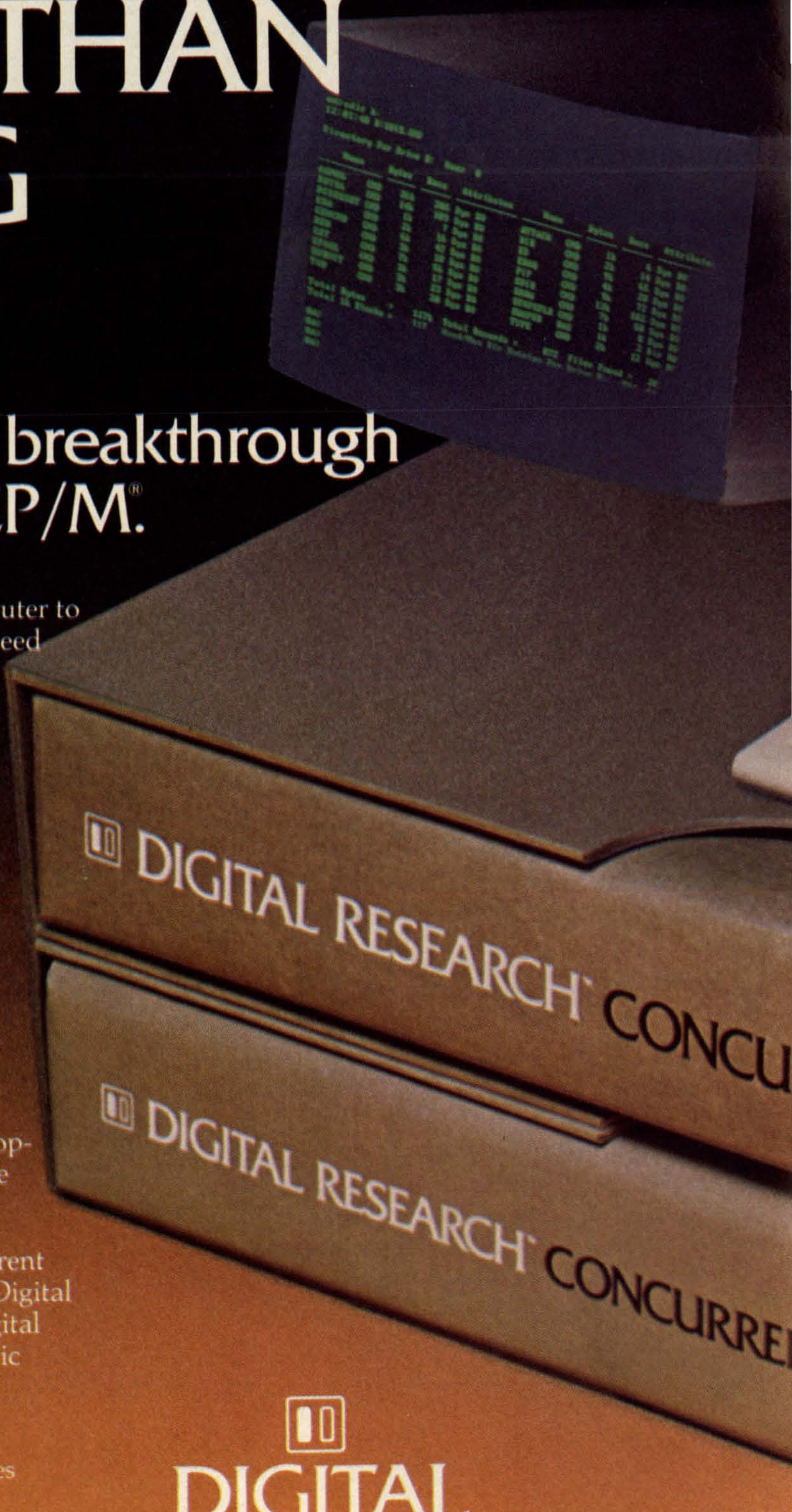
Get Input From File

CP/M Plus lets you get keyboard data from a file. This utility lets you design system operations to occur automatically, with a minimum of typing.

For example, you might want to start your modem program, provide necessary modem control commands, and log-on to The Source or CompuServe all in one command. You could do this by putting the necessary modem control commands and log-on information (name, password, etc.) into a text file, and using the GET utility just before starting the modem program.

CONCURRENT CP/M-86™ LETS YOUR COMPUTER DO MORE THAN ONE THING AT A TIME.

Another productivity breakthrough from the creators of CP/M®.



If you have to wait impatiently for your microcomputer to finish a job before moving on to another task, you need Concurrent CP/M-86. Digital Research introduces new software technology to increase the productivity of 16-bit microcomputers, including IBM's Personal Computer.

Using Concurrent CP/M-86, you can run several programs simultaneously, switching instantly from one program to another. For the first time you can write text while printing other documents. For the first time you can see directories without leaving your application program. For the first time you can edit programs while your computer compiles. Concurrent CP/M-86 is the best investment you can make in microcomputing, because it multiplies the value of your hardware, it lets you use all the CP/M-86 compatible programs, and if you're developing software, it insures that you're on the crest of the hottest new wave in the business.

As the chart shows, there's nothing else like Concurrent CP/M-86 in the microcomputing world, and only Digital Research offers it. For information, call or write Digital Research Inc., P.O. Box 579, 160 Central Ave. Pacific Grove, California 93950 (408) 649-3896.

Europe
Vector International
Research Park
B-3030 Leuven
Belgium
32 (16) 20-24-96
Telex 26202 VECTOR

Far East
Microsoftware Associates
6 Floor A. Y. Building
3-2-2 Kitaaoyama,
Minato-ku
Tokyo 107, Japan
03-403-2120
Telex 2426875 MSA

DIGITAL
RESEARCH™
The creators of CP/M.™

Circle reader service number 6

RENT CP/M-86™ IBM Personal Computer
T CP/M-86™

	CONCURRENT CP/M-86	CP/M-86	IBM PC DOS	UNIX™	iRMX-86™
Available on IBM Personal Computer	●	●	●		
Single-User with Background Consoles	●				
Real-Time Control	●				●
Password Protection	●			●	●
Time and Date Stamps	●		●	●	●
Record and File Locking	●				●
Local Area Network Support	●	●			●
Shipped with Editor, Assembler and Debugger	●	●			
1 Megabyte Memory Management	●	●		●	●
Winchester Disk Support	●	●		●	●
CP/M File Compatible	●	●			

ADDITIONAL FEATURES:

- 32 Megabyte File Capacity
- 512 Megabyte Disk Capacity
- Real-time message passing nucleus essential for multi-tasking, data communications and industrial control applications.
- Supported by existing CP/M-86 compatible languages.
- Upward compatible with MP/M-86.™

CP/M is a registered trademark of Digital Research Inc.
CP/M-86, Concurrent CP/M-86, and the Digital Research logo are trademarks of Digital Research Inc.
IBM PC DOS is a trade name of International Business Machines, Corp.
UNIX is a trademark of Bell Laboratories.
iRMX-86 is a trademark of Intel Corporation.

© Copyright 1982 by Digital Research

```
GET FILE filespec↵
GET FILE filespec [option] ↵
GET CONSOLE↵
```

Gets keyboard data from the file you specify, or gets it directly from the keyboard (console). With a *GET FILE* in action, you do not have to type at the keyboard — console input to your program comes directly from the file until the program ends or the file's data is used up (or a *GET CONSOLE* command in the file sends it back to the console for input). The options are:

SYSTEM — all input comes from the disk file specified in the command line. GET takes system and program input from the file until the file is used up or until GET reads a *GET CONSOLE* command from the file.
ECHO — displays (echoes) all input on the screen. This is the default option.
NO ECHO — turns off echo.

Without the *SYSTEM* option, GET gets input from the file for the next program execution, and reverts back to getting input from the keyboard when the program terminates. With the *SYSTEM* option, all input comes from the file. To revert back to getting input from the keyboard, use the *GET CONSOLE* command in the command file. Otherwise, input reverts back to the keyboard when the command file data is used up. Examples:

```
A〉GET FILE LOG-ON.NUM↵
```

— tells system to activate GET, but since *SYSTEM* is not specified, system still reads the keyboard until you execute a program:

```
A〉MYPROG↵
```

— all subsequent input comes from the file LOG-ON.NUM. When MYPROG terminates, the system reverts back to the console for console input.

```
A〉GET FILE SEARCH1.T [SYSTEM] ↵
```

— tells system to immediately get all subsequent input from the file SEARCH1.T.

HELP Messages

CP/M Plus provides help messages for all commands and various topics and subtopics.

```
HELP↵
HELP topic↵
HELP topic [NOPAGE] ↵
HELP topic subtopic↵
```

Display a list of topics and summarized information for CP/M commands. HELP with a *topic* displays information about that topic; HELP with a topic followed by a *subtopic* displays information about that subtopic. The first two letters is usually enough to identify a topic. When HELP displays its prompt, type:

? — to display list of main topics
. — (period) to redisplay what you just read.
.subtopic — to read a subtopic help message.
↵ — (RETURN) to return to the CP/M system prompt.

In addition you can specify the *NOPAGE* option to disable the 24 lines per page display. The NOPAGE option can be used with any variation of HELP command. Here are examples of the HELP command:

```
A〉HELP DATE↵
A〉HELP DIR OPTIONS [N] ↵
```

— selects the topic DIR and subtopic OPTIONS with *NOPAGE* (abbreviated *N*) in effect.

Creating Your Own Help Messages

CP/M Plus provides the file HELP.HLP, to which you can add help messages to the HELP utility. You cannot edit HELP.HLP; you must first use the HELP command with the *[EXTRACT]* option to extract the messages from HELP.HLP into a new file, HELP.DAT. You can then edit HELP.DAT, change error messages, add messages, etc. Use HELP with the *[CREATE]* option to build a revised HELP.HLP file to use with the HELP utility.

To start the process, type:

```
A〉HELP [EXTRACT] ↵
```

(*[EXTRACT]* can be abbreviated to *[E]*.) This command creates HELP.DAT, which you then edit with your favorite word processing program.

To enter or change messages, you have to follow these rules:

- Topics must be in alphabetical order.
- Topic names start with three slashes (*///*), followed by a number, and the name, followed by RETURN:

  ```
  ///nTopicname↵
  ```

- Subtopics should be put underneath the major topic, and listed alphabetically, using the number *n* to indicate level: *1* is a major topic, *2* is a major subtopic, *3* is a minor subtopic, *4* is a sub-subtopic, etc., up to *9*. Here are examples:

  ```
  ///1ACCOUNTING↵
  ///2INVENTORY↵
  ///3INVOICES↵
  ///2PAYABLES↵
  ///2RECEIVABLES↵
  ///3REPORTS↵
  ///3STATEMENTS↵
  ```

The major topic is accounting, with major subtopics inventory, payables and receivables, and minor subtopics invoices, reports and statements.

After editing HELP.DAT, use the following command to rebuild the HELP.HLP file:

```
A)HELP [CREATE] ↩
```

(*[CREATE]* can be abbreviated to *[C]*.) Your additional or edited messages are now available through the HELP utility. You can abbreviate the topic and subtopic names to their first three letters when calling the messages from HELP.

PIP: Copying To/From Files, Disks, and Devices

The "familiar" PIP utility is enhanced to include an *A* (archive) parameter to copy all files on disk with the *ARCHIVE* attribute (files displayed by *DIR* that have not been "backed up" since the last access). In addition, PIP can now copy *to* a user area as well as *from* a user area. Note that PIP's performance is affected by the *SET* command's ability to change the *ARCHIVE* attribute and *PASSWORD* protection modes.

```
PIP filecopy= filespec↩
PIP filecopy= filespec [p] ↩
```

Make a copy of a file using another name for the copy. An intermediate file "*filecopy.$$$*" is created until the PIP operation ends by renaming the "*.$$$*" file to the destination file specified. Optional PIP parameters (listed below) can be used. Examples:

```
A)PIP NEWONE.TXT= OLDONE.TXT↩
```

— copies OLDONE.TXT to NEWONE.TXT in the current user area (0) on the current logged drive (drive A).

```
A)PIP B:NEWONE.TXT= OLDONE.TXT [Z] ↩
```

— uses the *Z* option to zero the parity bit while copying the file. Note use of *B:* to locate NEWONE.TXT on drive B.

In addition, CP/M Plus PIP lets you copy a file into another user area:

```
A)PIP FINAL.TXT [G1] = PRELIM.TXT↩
```

— copies PRELIM.TXT to FINAL.TXT in user area 1.

```
PIP filecopy= filespec, filespec↩
PIP filecopy= filespec [p] , filespec [p] ,... ↩
```

Concatenate one or more files together into a new file by another name (*filecopy*), appending the last *filespec* to the end of the *filespec* to its left, and so on. Optional PIP parameters are available for each *filespec*. Example:

```
A)PIP B:RESULT.TXT= TEST1,B:TESTADD [Z] ,TEST2,C:TEST3↩
```

— concatenate the files TEST1 from drive A, TESTADD from drive B (and zero the parity bit in this file), TEST2 from drive A and TEST3 from drive C into the file RESULT.TXT on drive B.

```
PIP d:= filespec↩
PIP d:= filespec [p] ↩
PIP d:= filespec [p] ,filespec [p] ,... ↩
```

Make a copy with an identical name on another disk, or concatenate each *filespec* into a file using the name of the first *filespec*, and possibly use parameters. Example:

```
A)PIP B:= RESULT.TXT,B:TEST4 [Z] ,C:TEST5↩
```

— concatenate RESULT.TXT on drive A with TEST4 on drive B and TEST5 on drive C to a new version of RESULT.TXT on drive B.

```
A)B: [G1] = A:MOM.TXT↩
```

— copies MOM.TXT to drive B, user area 1.

```
PIP d:= d:filematch↩
PIP d:= d:filematch [p] ↩
```

Copy several files at once using a filename match, with identical filenames used for the copies on another disk. Example:

```
A)PIP B:= A:*.TXT↩
```

— copies all ".TXT" files on drive A to drive B.

```
PIP dev:= filespec↩
PIP dev:= filespec [p] ↩
PIP dev:= filespec [p] ,filespec [p] ,... ↩
PIP dev:= d:filematch [p] ↩
```

Copy one or more files to an output device. You can use optional PIP parameters. Example:

```
A)PIP LST:= KELLY.DAD [P55] ↩
```

— copies KELLY.DAD to the LST: device (printer), causing a Form Feed to occur every 55 lines.

```
PIP filespec= dev:↩
PIP filespec [p] = dev:↩
```

Copy input from device to a file, using optional parameters.

PIP Device Names

CON:	Console (input is keyboard, output is screen)
RDR:	Reader device (input)
PUN:	Punch device (output)
LST:	Printer device (output)
NUL:	Send 40 "nulls" (0 HEX)
EOF:	Send "end of file" (↑Z)
OUT:	Custom output device
INP:	Custom input device
PRN:	A special form of LST: used to expand tabs (every eighth column), number lines, and eject pages (every 60 lines).

PIP Parameters

Parameters must be enclosed in square brackets. You may separate several options with blanks (e.g., *[F P55]*).

A	Archive (copy) files that have changed since the last copy, or that have the archive attribute *SET* to off (*[ARCHIVE= OFF]* option of *SET*). Type the *DIR* command to display the archive attribute setting. The *[A]* PIP parameter turns the archive

	attribute on for each file copied.
B	Block mode transfer. **CP/M 2.2 only**.
C	Confirm. PIP prompts for confirmation before each file copy. Use *[C]* when you don't want to copy all files in a specified filematch.
Dn	Delete characters after *n*th column. Use to print or display uncommented listings when comments begin after a specific column number (c). Use (c-1) for *n* to delete comments from a printed listing or display(e.g. *[D79]* to delete comments beginning in column 80).
E	Echo copy operation at console.
F	Filter out form feeds. Use in conjunction with *[Pn]* parameter to position page breaks and insert form feeds only after a specified number of lines (e.g. *[F P62]* to ignore any imbedded form feeds and insert a form feed after every 62 lines copied).
Gn	Get file(s) from or copy file(s) to user area *n* (an integer 0-15).
H	Check for proper Intel hexadecimal format and display error messages when errors are found during HEX copy operation.
I	Ignore null records (:00 Intel Hexadecimal file format) during copy operation. This option automatically sets *[H]* option.
L	Convert UPPER to lower case letters during copy.
N	Add line numbers to each line copied, starting at 1 and incrementing by 1 with a colon ending each line number. Specify *[N2]* to add leading zeros and insert a tab following the line number. Tab length can be expanded with use of the *[T]* parameter.
O	Ignore ↑*Z* end-of-file marker. Used for concatenating object code files which have imbedded end-of-file markers that would otherwise end the PIP operation prematurely.
Pn	Add a form feed after *n*th line. A form feed is added after 60 lines if *n* is not specified or *n* is 1.
Qstring↑*Z*	Copy portion, stopping on *string*.
R	Copy system files.
Sstring↑*Z*	Copy portion, starting on *string*.
Tn	Set tab stops every *n*th column.
U	Convert lower to UPPER case letters during copy.
V	Verify copy.
W	Write a new file (and its attributes) over a read-only file, without asking for confirmation.
Z	Set parity bit to 0.

Sending Output To a File

CP/M Plus provides the PUT utility to let you send both console output and printer output to a file rather than the console or printer. This can be very useful, especially if the output of one program can be used as the input to another (using GET). Another use is to send printer output to a file when you have no working printer. PUT echoes to the console, unless you specify the no-echo option. By sending console output to a file, you can capture menu screens from programs and edit them into help messages. Use HELP to add them to the HELP facility.

```
PUT CONSOLE FILE filespec↵
PUT CONSOLE FILE filespec [option list] ↵
PUT CONSOLE CONSOLE↵
```

Put sends display (console) output from the next program into the file you specify until the program terminates; or if you use the *SYSTEM* option, sends all output immediately into the file until you type the *PUT CONSOLE CONSOLE* command to redirect display output back to the console. *PUT CONSOLE* is an abbreviation of *PUT CONSOLE OUTPUT TO.* Options are:

ECHO — Displays (echoes) all output on the screen. This is the default option.
NO ECHO — Turns off echo.
FILTER — Filters all control characters by translating them to printable ASCII characters (e.g., ESC translates to ↑[).
NO FILTER — Does not filter control characters. This is the default option.
SYSTEM — Puts all output immediately to the disk file specified in the command line. PUT directs system and program output to the file until you supply the *PUT CONSOLE CONSOLE* command to redirect output to the display.

Examples:

```
A>PUT CONSOLE FILE RESULT.TXT↵
```

— tells system to activate PUT CONSOLE, but since *[SYSTEM]* is not specified, output is sent to the display until you execute a program:

```
A>MYPROG↵
```

— sends all subsequent output to the file RESULT.TXT. When MYPROG terminates, the system reverts back to sending output to the display (console).

```
A>PUT CONSOLE FILE RESULT.TXT [SYSTEM] ↵
```

— sends all console output immediately into the file RESULT.TXT, until a *PUT CONSOLE* command redirects output back to the display (console).

```
A>PUT CONSOLE FILE B:RESULT.TXT [NO ECHO,FILTER] ↵
```

— puts console output into B:RESULT.TXT, with control character filtering and no echoing of the output on the screen.

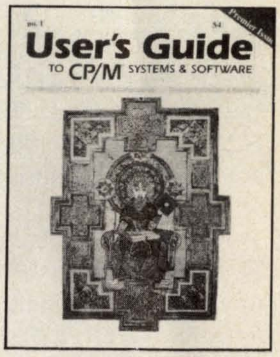
```
A>PUT CONSOLE CONSOLE↵
```

— redirects console output to the console display.

```
PUT PRINTER FILE filespec↵
PUT PRINTER FILE filespec [option list] ↵
PUT PRINTER PRINTER↵
```

FREE INFORMATION
User's Guide Reader Service Card

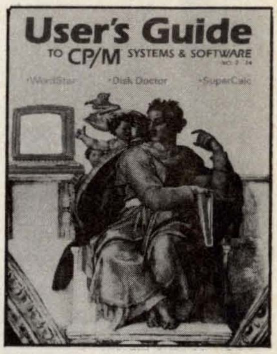
User's Guide
P.O. BOX 3050
STANFORD, CA 94305



No. 1
$4
Premier Issue
User's Guide
TO CP/M SYSTEMS & SOFTWARE

Each advertisement and product announcement in this issue has a reader service number. To receive by mail from the manufacturer, simply circle the same number on this reader service card, and drop the card in the mail. Your name and address is sent on to the appropriate manufacturer, who sends the information.

The service helps you get detailed information about products you may want to purchase. *User's Guide* does not sell or recommend products associated with this service.



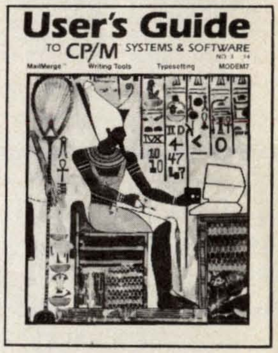
User's Guide
TO CP/M SYSTEMS & SOFTWARE
WordStar
Disk Doctor
SuperCalc

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25
26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50
51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75
76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 100

Please take the time to fill out:

1. How would you classify your experience with computers?
   ☐ Novice ☐ Intermediate ☐ Advanced
2. Which computer do you use? ____________________
3. Please describe your system. ____________________
   Memory ____________________
   Printer ____________________
   Monitor ____________________
   Storage devices ____________________
   Other Components ____________________
4. Name some applications programs that you use most with your computer.
   ____
5. Where do you use your computer?
   ☐ Home ☐ School ☐ Small-large business environment
   ☐ Large-business/Corporate environment ☐ Other ____________________

Name ____________________ Age ________ Sex ________
Occupation ____________ Company ____________ Title ____________
Address ____________ City/State/Zip ____________



User's Guide
TO CP/M SYSTEMS & SOFTWARE
MailMerge
Writing Tools
Typesetting
MODEM7

Mail To: P. O. Box 3050, Stanford, CA 94305

Advertiser Index

Service		Page
1	Alphamerics	79
2	Alspa	C4
3	CompuPro	C2
4	Computer Thaumaturgy	77
5	Computing!	C3
6	Digital Research	42
7	Hawkeye Grafix	35
8	Heritage Software	27
9	KIAI Systems	21
10	Omni Systems Intl.	35
11	Paradigm Consultants	51
12	Reston Publishing	19
1	User Friendly Software	79
13	Wizard of OsZ	59

Puts printer output into the file you specify until the program terminates; or if you use the *SYSTEM* option, puts output into the file until you type the *PUT PRINTER PRINTER* command to redirect printer output back to the printer. *PUT PRINTER* is an abbreviation of *PUT PRINTER OUTPUT TO.* Options are the same as above for *PUT CONSOLE*, except that *NO ECHO* is the default setting, not *ECHO*. Examples:

```
A>PUT PRINTER FILE RESULT.TXT↵
```

— tells system to activate PUT PRINTER, but since *SYSTEM* is not specified, system still sends printer output to the printer until you execute a program:

```
A>MYPROG↵
```

— sends all subsequent output to the file RESULT.TXT. When MYPROG terminates, the system reverts back to sending printer output to the printer.

```
A>PUT PRINTER FILE RESULT2.TXT [ECHO,SYSTEM] ↵
```

— sends all printer output to the file RESULT2.TXT as well as to the printer (using *ECHO* option), and the PUT remains in effect until you enter a *PUT PRINTER PRINTER* command.

```
A>PUT PRINTER PRINTER↵
```

— sends subsequent printer output to the printer.

Renaming Files

The CP/M Plus RENAME (or REN) command is enhanced to allow you to specify a filename match (such as *A??.**) to rename several files at once.

```
RENAME newname=oldname↵
RENAME b:newname=oldname↵
RENAME newmatch=oldmatch↵
RENAME b:newmatch=oldmatch↵
```

Changes the name of a file on the current disk, or on another disk. Specify the other disk with the *newname* (RENAME does not copy files or transfer files to another disk.) To change several filenames in one command use the * and/or ? "wildcards" in a filename match. The RENAME command can be abbreviated REN. *REN* (or *RENAME*) by itself displays prompt messages for the old and new names. Examples:

```
A>RENAME↵
```

— system will prompt you for new and old names. A "No File" message is displayed if old name can't be found, or an existing file with same name can be overwritten if you answer *Y* to delete in answer to "Not renamed" message.

```
A>RENAME B:NEWFILE.LST=OLDFILE.LST↵
A>REN NEWFILE.LST=B:OLDFILE.LST↵
B>REN NEWFILE.LST=OLDFILE.LST↵
```

— renames OLDFILE.LST on drive B to NEWFILE.LST.

```
A>REN S*.TEX=A*.TEX↵
```

— renames all files on drive A that start with "A" and end with ".TEX" to names that start with "S" and end with ".TEX".

File/Disk Access and Passwords

Access to disks and files can be controlled by passwords and attributes if you wish. There is no need to use passwords unless you need to protect your files and disks from accidental erasures. If you use password protection, write passwords down as you assign them. You lose access to files and disks protected by passwords you can't remember. The CP/M Plus *ARCHIVE* option lets you assign files to the automatic backup feature of PIP.

Set File Attributes

```
SET filespec [option list] ↵
```

Sets the file access attributes to *RW* (read-write), *RO* (read-only), *DIR* (directory display), *SYS* (system file), and set other attributes. File attributes include:

DIR — changes a system file (*SYS*) to a directory file.
SYS — changes a directory file (*DIR*) to a system file.
RO — changes a read-write enabled file (*RW*) to a read-only file.
RW — changes a read-only file (*RO*) to a read and write enabled (changeable) file.
ARCHIVE=OFF — sets the archive attribute to "off," marking the file ready for backup using the PIP command with the *A* option. PIP sets the archive attribute "on" after copying the file.
ARCHIVE=ON — sets the archive attribute to "on," so the file will not be copied in your next PIP operation using the *A* option.
F1=ON or *OFF* — turns on or off a custom file attribute. You can have attributes *F1* to *F4*.

Set Drive Attribute

```
SET d: RO↵
SET d: RW↵
```

Sets the disk drive attribute to read-only (*RO*) or read-write (*RW*). Drives are usually read-write. If a drive is read-only, you cannot rename, erase or change files, nor can you copy to the disk until a Control-C (↑*C*) resets all drives to read-write.

Set Disk Label and Password (Banked Version Only)

```
SET [NAME=diskname] ↵
SET d: [NAME=diskname] ↵
SET [PASSWORD=password] ↵
SET [PASSWORD= ↵
```

Labels a disk and protects the label with a password of up to eight characters. To turn off the password, use the form *SET [PASSWORD=* ↵. The *diskname* resembles a filename with extension (the default name is LABEL). If you do not password-protect the disk label, anyone with access to the disk can password-protect it from your access. SET always asks for the password of a protected disk label. If you use a disk password, be sure to write it down; if you forget it, you will lose access to your disk. Examples:

```
A>SET [NAME=WSBOOK] ↵
```

— sets the drive A disk label to WSBOOK.

```
A>SET [PASSWORD=DRAFT1] ↵
```

— sets the drive A disk password to DRAFT1.

Set Time Stamp Options

```
SET [CREATE=ON] ↵
SET [ACCESS=ON] ↵
SET [UPDATE=ON] ↵
```

Keeps a record of the time and date of file creation or last access, with a record of the last update of the file. To enable time stamping of files, you must first run *INITDIR* to format the disk directory. Options mean:

[CREATE=ON] — keeps track of creation time for all files on the current drive. To record the creation time of a file, you must turn on this option for the disk before you create the file. To keep track of the last update of each file as well, include the *[UPDATE]* option.

[ACCESS=ON] — keeps track of the last access (edit or display, i.e. through ED or TYPE commands) of all files on the current drive. Access and create options are mutually exclusive; only one can be in effect at a time. When you turn on the access option, the create option is automatically turned off. To keep track of the last update (edit or change) to the file, include the *[UPDATE]* option.

[UPDATE=ON] — keeps track of the last update to all files on the current drive. The update option records the time each file was last modified.

The DIR command with the *[FULL]* option displays the dates for each file — create and update, or access and update. Examples of the SET command:

```
A>SET [ACCESS=ON] ↵
A>SET [CREATE=ON,UPDATE=ON] ↵
```

Turn On/Off Password Protection (Banked Version Only)

```
SET [PROTECT=ON] ↵
SET [PROTECT=OFF] ↵
```

Turns on or off file password protection for an entire disk. Password protection must be on to assign passwords to files.

Assign Passwords to Files (Banked Version Only)

Passwords are required for access, unless you set a default (*[DEFAULT=]* option) password for a type of access (such as read-only) that lets system operators access files without knowing passwords.

Always write down your passwords — if you forget a file's password, you cannot use the file unless you turn off password protection for the entire disk (for which you need the disk's password). If you forget a disk's password, you cannot turn off password protection for the disk.

```
SET [DEFAULT=password] ↵
SET filespec [PASSWORD=password] ↵
SET filespec [PASSWORD=password,PROTECT=mode] ↵
SET filespec [PROTECT=mode] ↵
SET filespec [PROTECT=NONE] ↵
```

Sets a password to one or more files, or sets a default password for use with all files during system operation. You can use a filename or filename match for *filespec*. Passwords can be up to eight characters (lower case letters are translated to all upper case). Protection *modes* of access are as follows:

READ — password is required for reading, copying, printing, writing, deleting or renaming the file.

WRITE — password is required for writing, deleting or renaming the file. You do not need a password to read, copy or print the file.

DELETE — password is required only for deleting or renaming the file. You do not need a password to read or modify the file.

NONE — if a password exists, use this option to delete it.

Here are examples of file passwords:

```
A>SET TONY2.TXT [PASSWORD=TUGSTUFF] ↵
```

— assigns the password TUGSTUFF to the file TONY2.TXT.

```
A>SET B:*.TXT [PASS=USERGUID,PROT=WRITE] ↵
```

— assigns the password USERGUID to all the files with the ".TXT" extension on drive B. Each file is protected against writes to prevent unauthorized editing, changing, deleting or renaming. No password is required for reading, copying, or printing the files.

```
A>SET [DEFAULT=COMMON] ↵
```

— sets COMMON as the default system password while the system is up and running. The system uses COMMON to access files if you do not type a password for a password-protected file. If the file's password is COMMON for a certain mode of access, you can access the file without specifying the password.

```
A>SET *.COM   [SYS,RO,PASS=KNOWIT,
PROT=READ] ↵
```

— sets the most protection for all the programs (".COM" files) on the disk in drive A: protection against reading or using in any way unless you know the password KNOWIT (unless the default password is set to KNOWIT). Even if you know KNOWIT, you cannot write to the file until you change its *RO* (read-only) attribute. The *SYS* attribute keeps the files hidden (not listed in a normal DIR display).

```
A>SET *.COM [DIR, RW, PROTECT=NONE] ↩
```

— removes the protection set in the previous example: no password protection on the file, the *RW* (read-write) attribute, and the *DIR* attribute for displaying the files in the directory.

Define Program Searching

SETDEF lets you define the order in which disk drives are searched for ".COM" (program) files and ".SUB" (SUBMIT) files. The system usually searches the drive specified in a *filespec*, or the current drive. SETDEF can extend this search to include other drives. The search only works when calling ".COM" files for execution, or submitting a ".SUB" file for execution.

```
SETDEF↩
```

SETDEF by itself displays the current order of disk drive searching, the temporary drive, and the current order of filename extension (file type) searching.

```
SETDEF d:,d:,d:,d: ↩
SETDEF d:,*,d:,d: ↩
SETDEF [TEMPORARY=d:] ↩
SETDEF [ORDER=(ext)] ↩
SETDEF d:,d:,*,d: [TEMP=d:,ORDER=(ext,ext)] ↩
```

Lets you define up to four disk drives in the order they should be searched for programs or SUBMIT files to execute, and lets you define the drive to hold temporary files, and the file extension search order. The SETDEF definitions affect only the loading and execution of ".SUB" files. Examples:

```
A>SETDEF [TEMPORARY=B:] ↩
```

— sets disk drive B as the drive to be used for temporary files.

```
C>SETDEF B:,* ↩
```

— tells the system to search for a program on drive B, then, if not found, search for it on the current drive (drive C in this example).

```
A>SETDEF [ORDER=(COM, SUB)]
```

— tells the system to search for a ".COM" file to execute on the current drive. If no ".COM" version of the file is found, search for a ".SUB" file. The default search order is ".COM" files only.

Define Display Modes

```
SETDEF [DISPLAY] ↩
SETDEF [NO DISPLAY] ↩
SETDEF [PAGE] ↩
SETDEF [NOPAGE] ↩
```

Turns on/off the system display and screen page modes. When display mode is on, the system displays the location and name of programs loaded or ".SUB" files executed. In page mode, the CP/M Plus utility programs pause after displaying one full screen of information and don't resume until you press a key on your keyboard.

Display Disk Information

```
SHOW↩
SHOW d:↩
SHOW d: [option list] ↩
```

Displays the following disk drive information:

— Access mode and the amount of free disk space
— Disk label
— Current user number and
— Number of files for each user number on the disk
— Number of free directory entries for the disk
— Drive characteristics.

Options are:

SPACE — access mode and disk space.
LABEL — disk label and time stamp.
USERS — current user area number, and all user areas on the drive with corresponding number of files assigned to them.
DIR — number of free directory entries.
DRIVE — detailed drive characteristics.

Examples:

```
A>SHOW [SPACE] ↩
```

— displays access mode and amount of space left on logged-in drives.

```
A>SHOW B: [LABEL] ↩
```

— displays label information for drive B.

Submitting a Group of Commands

The SUBMIT utility lets you execute a group of commands (one after another) line-by-line as if you were typing them. You type the commands into a text file with the ".SUB" extension, and use SUBMIT to execute the ".SUB" file. SUBMIT lets you include variables in the ".SUB" file for which you can assign words or partial words ("arguments") in the SUBMIT command line.

The ".SUB" file can contain the following types of lines:

- Any valid CP/M command
- Any valid CP/M command with SUBMIT variables (*$0-$9*)
- Any data input line
- Any program input line with variables (*$0* to *$9*).

Note however, that the command line cannot exceed 135 characters. The commands can include *$1*, *$2*, up to *$9* variables, for which arguments are assigned based on the order they appear in the SUBMIT command line.

The following example includes samples of lines which may be entered in a ".SUB" file (this one's called WRAPUP.SUB):

```
DIR $1.BAK↵
ERASE $1.BAK [CONFIRM] ↵
PIP↵
<B:=$1.TXT↵
<CON:=$1.TXT↵
<↵
DIR B: ↵
```

The argument *SAMPLE* could be substituted for the variable *$1* in this ".SUB" file with the following command:

A>*SUBMIT WRAPUP SAMPLE*↵

This would cause SAMPLE.BAK to be erased (with a confirmation), and SAMPLE.TXT to be copied to drive B and to the console display.

SUBMIT↵
SUBMIT filespec argument argument... ↵

Executes a group of commands in a ".SUB" file. *SUBMIT* by itself asks for the name of the ".SUB" file, and you type its name and any arguments you need to supply for variables in the file.

Automatic Program Startup

Every time you power up or reset your computer, CP/M Plus looks for a special submit file named PROFILE.SUB to execute. If PROFILE.SUB does not exist, CP/M Plus resumes normal operation. If the PROFILE.SUB file is found, the system executes the commands in the file. This file is convenient to use if you regularly run commands or utility programs every time you start the system, or if you want to implement a simple menu-driven operation.

Display File Contents

TYPE filespec↵
TYPE filespec [PAGE] ↵
TYPE filespec [NOPAGE] ↵

Displays the contents of a text file (ASCII characters) on your screen. *[PAGE]* displays contents in paged mode — the system pauses after listing a screen-full of text (usually 24 lines) until you hit a key to continue. *[NOPAGE]* turns off page mode; displays are continuous without any pauses.

Set User Area

When you have a large amount of disk space for a given drive, you may want to separate files by placing them in "user areas" identifiable by number (user area 0, 1, etc.). The disk is not actually carved into areas; the numbers are added to the file's description in the directory, and you need to remember the user area to find the file.

For example, your text files may be in user area 3; if you are in user area 1, you won't see them until you issue the USER command to change the user area to 3. System files and programs are usually placed in user area 0 (a "common" area) with the "SYS" attribute, where they can be found from other user areas.

USER↵
USER number↵

Sets the current user area to *number*. The disk directory can be divided into distinct groups according to user area numbers. User area numbers range from 0 through 15. The user area, if not 0, is displayed in the CP/M prompt (e.g., 4B> means the current user area is 4, and the current drive is drive B). Examples:

B>*USER*↵
Enter User*4*↵
4B>

— changes the current user area from 0 to 4 on drive B.

A>*USER 3*↵
3A>

— changes the current user area to 3.

LOOK!
I don't want all that!

If you feel the same way about large, complex accounting packages then you should try MINI-LEDGER, our single-entry program for people who want results, not complications.

- 99 user-definable payment and income codes
- Provides almost all the information needed for taxes and financial statements
- Virtually instant analysis of your cash-flow for any period from 1 to 12 months
- You don't have to be an accountant to use it

Requires: CP/M™ 64 K RAM and at least one disk drive

Formats: IBM 8" Single-Density - Eagle
Osborne Single-Density - Superbrain

Cost: $150 complete program
$ 30 manual-only with demo disk
California Customers add 6.5% sales tax

To Order: Call or Write
Paradigm Consultants
39812 Mission Blvd. (203)
Fremont, CA 94539
(415) 657-4095

MINI-LEDGER

CP/M is a Registered Trademark of Digital Research Corporation

Circle reader service number 11

NEC PC-8201
LOW BATTERY

Continued from page 36

gramming utilities, Access Manager and Display Manager. The Pascal/MT+ pack also includes Access and Display Manager. These packs are priced to be much less than the sum of its parts.

DRI announced an agreement with VisiCorp to produce a version of the recently announced VisiOn® package for CP/M computers. DRI also announced a new Personal Basic that executes MBASIC programs. Personal BASIC is available for CP/M-86, Concurrent CP/M-86 and MP/M-86.

Meanwhile, Hewlett-Packard signed a licensing agreement with DRI to put the new CP/M-68 system on the HP Series 200 personal computers. CP/M-68 is designed for the 68000 chip (a high performance 16-bit/32-bit chip), and it provides a path for transporting programs and data to and from the UNIX world. List price is $350, which includes a C compiler and run-time library compatible with UNIX version 7.

Xerox joined the pack in providing both CP/M-86 and CP/M Plus for its 16/8 model "Professional" computer. This computer offers both an eight-bit Z80A chip and an 8086 chip — two processors which can run simultaneously, letting you execute two programs at once. Xerox 820-II users can upgrade their units, and software and data is compatible with the 16/8 system since both run CP/M.

CompuPro announced a 68000-based system with CP/M-68K®, the System 816 Model 68K, using CompuPro's 8MHz 68000 CPU board with 256K of internal memory expandable to 1M, three serial ports, one parallel and one Centronics/Epson port, and M-DRIVE/H®, 1.5M of RAM used as a memory disk. Price: $8995.

With CP/M-68K, applications written in the C language under UNIX can be transported into the CP/M environment, using the standard CP/M file format. CompuPro offers the complete line of DRI programming languages and productivity tools.

Large volume distributors of CP/M systems now include DEC, NCR, Texas Instruments, Hewlett-Packard, Tandy/Radio Shack, Xerox, Sony, Fujitsu, NEC, Sanyo, Vector Graphic, Corona, CompuPro and more.

Circle number 51 on the reader service card.

New Portables:

NEC PC-8200

NEC Home Electronics introduced a very compact battery-operated computer with 16K internal RAM (read-write memory) expandable to 64K with an additional RAM cartridge, and 32K ROM (read-only memory) of built-in software. The software lets users do text writing, editing and telecommunicating to other computers. A built-in BASIC language is also supplied for programming.

The unit weighs only 3.8 pounds, and comes with a built-in nickel cadmium battery that always provides low-level power to the memory. The

Continued on page 78

A Reference Guide to DU (Disk Utility)

An "Impatient User's Guide" to DU (Disk Utility), a famous disk fixer from the public domain libraries

(adapted from Ward Christensen's DU documentation)
by Kelly Smith

DU is a comprehensive utility that allows a user to examine and modify *any portion* of a disk used with CP/M-80 (Version 2.0 or greater).

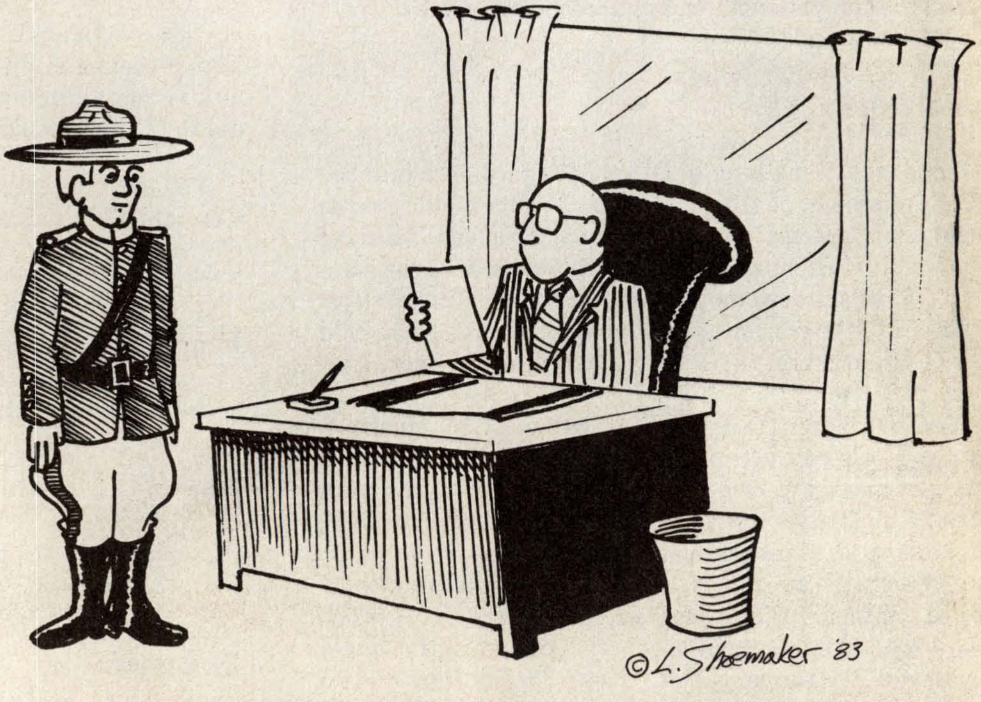
DU is an invaluable tool for recovering disks with "bombed" directories, or for modifying CP/M "run time" parameters, or for recovering accidentally-erased files, or even for making a backup copy of your disk's directory and preserving it in a safer place on your disk. The possibilities are virtually endless.

DU lets you change actual bytes on the disk. Such a direct method may seem logical to an assembly language programmer, but for users who are unfamiliar with bytes and hexadecimal numbers, DU can not only be hard to learn, but dangerous!

This reference guide does not try to be tutorial (one is coming in a future issue — *the editors*), but it should be useful to anyone who has tried to use a disk-fixing program. To understand disk directories, you should read "BIOS, Files and the Directory" in *User's Guide* #2.

DU provides a "help" facility — just type a question mark (?) followed by RETURN, to see a summary of the extensive command set (most DU commands are single-key commands you type and end with RETURN).

A word of caution to the novice user: it is possible to DU a



"Your resume is very impressive... but we advertised for someone with RCPM experience."

disk to death! That is, casual modification of portions of the CP/M-80 operating system data tracks is usually destructive to the proper operation of your system disk. Use particular care when writing your modifications back to disk!

The best reason for using DU is that it is *free*. The program and its documentation were written by Ward Christensen, who must hold the world's record for producing public domain programs. (Thanks, Ward, for enriching the CP/M world with these wonders!) Ward included the "source code" for the assembly language hackers out there who want to customize everything they can get their hands on... Remember, if you customize software, you're "on your own" — don't expect "hand-holding" from the original author when you write on their wall! Please give credit where it is due so we can all follow the evolution of public domain software.

Newcomers and non-programmers: just because we say it's dangerous and hard to use, *don't* forget about DU. It is free and available from most user groups (and there's usually a DU hacker lurking in the group). DU is functionally equivalent to some commercial programs, so before you buy a disk utility program, check out DU — you won't find a better program for your money!

Typeface Conventions

In this reference guide we use *THIS typeface* to show commands and expressions you type at your keyboard, and THIS typeface to show displayed messages.

In addition, we use these symbols:

↵	RETURN key (CR or ENTER on some keyboards)
↑	Control key (CNTL, CTRL or ALT on some keyboards)

Also, where indicated in the command summary, we use *italics* to show optional items in a command's syntax.

Installation

Versions 7.0 and later of DU are designed to be installed with a minimum of trouble. In fact, in almost all cases, no changes are necessary to get DU up and running. For those who want to customize DU, and for those who have non-standard CP/M systems, the 8080 source code is available from various RCPM systems as well as directly from the CP/M User's Group library (CPMUG, 1651 Third Avenue, New York. N.Y. 10028).

The only requirement is that you must be running a CP/M-80 system, version 2.0 (or greater). This is because DU uses the disk parameter block incorporated in the latest releases of Digital Research's CP/M-80 to determine the characteristics of the disk environment.

The only parameter you might want to change (not required), is the CPU clock speed flag at address 103H (when DU is loaded into system memory). Leave this byte zero if you have a 2 MHz (megahertz) clock. Change it to 0FFH if you have a 4 MHz clock (refer to your Digital Research DDT documentation if you are unfamiliar with modification of program files, and patching techniques). This modification is needed only to use the *Z* (sleep) command — a command that lets you place a pause in a stream of commands to, for example, pause every 24 lines to read long displays. An alternative to modifying the program: use larger than usual numbers with the *Z* command when running DU with a computer system clock rate of 4MHz or greater.

Usage

The simple way to run DU is to type the program name from CP/M:

```
A⟩DU↵
```

Once DU is running, it will display a colon (:) as the DU command level prompt. DU expects single-letter commands followed by RETURN. To specify several commands at once, place them on one line and separate them with semicolons (;). You can also repeat the execution of a given command or string of commands a given number of times, or indefinitely (until you type a Control-*C*, shown from now on as ↑*C*).

If you want to perform a few operations in "expert mode," you can put DU commands on the CP/M command line with *DU*. For example, you can type:

```
A⟩DU G0;D;G2;=OK⟨D⟩⟨A⟩⟨1A⟩;D↵
```

That's a complex example, but it shows that you can type many commands at once, and execute them without further ado. Here's an example with an explanation:

```
A⟩DU M;X↵
```

This command runs DU and executes DU's *M* command. The *M* command displays a disk *map* — a display of the disk's group allocations. Figure 1 shows a disk map of a CompuPro 8/16 System disk.

You must type the *X* command to leave DU (if you type a Control-*C* (↑*C*) accidentally or on purpose, you will not "drop out" of DU).

Function Summary

All DU commands end with RETURN.

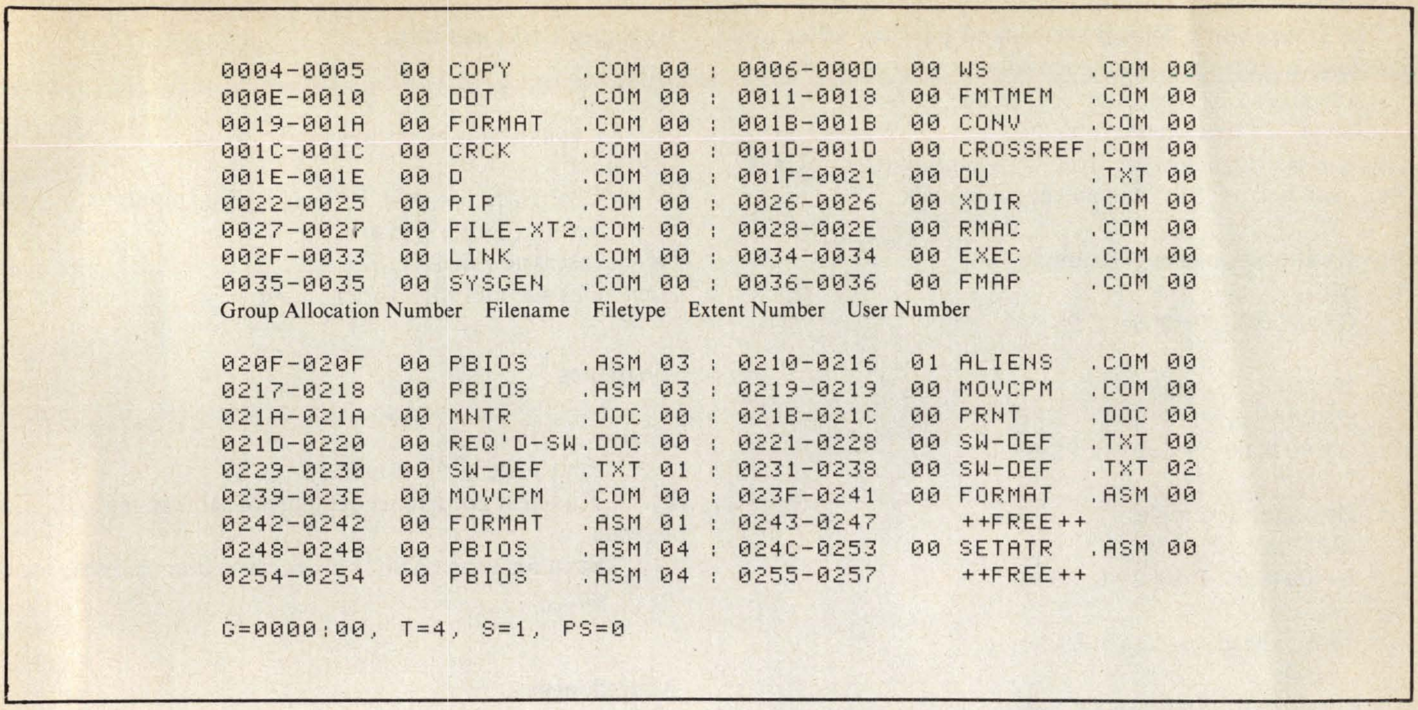
Help

? Request help. Example:

```
:?↵
```

Positioning

The only way to know where you are is to understand the

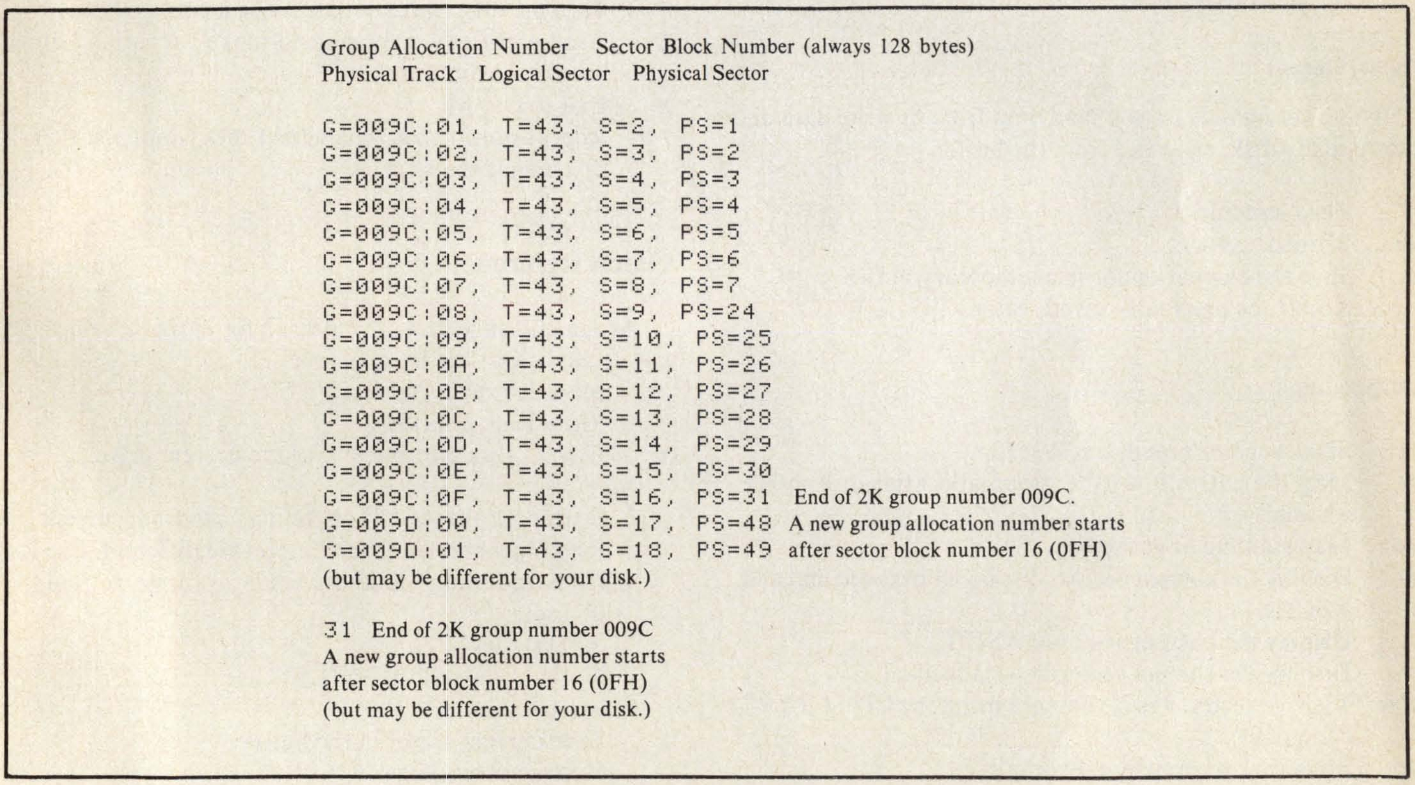


```
0004-0005   00 COPY     .COM 00 : 0006-000D   00 WS       .COM 00
000E-0010   00 DDT      .COM 00 : 0011-0018   00 FMTMEM   .COM 00
0019-001A   00 FORMAT   .COM 00 : 001B-001B   00 CONV     .COM 00
001C-001C   00 CRCK     .COM 00 : 001D-001D   00 CROSSREF.COM 00
001E-001E   00 D        .COM 00 : 001F-0021   00 DU       .TXT 00
0022-0025   00 PIP      .COM 00 : 0026-0026   00 XDIR     .COM 00
0027-0027   00 FILE-XT2.COM 00 : 0028-002E   00 RMAC     .COM 00
002F-0033   00 LINK     .COM 00 : 0034-0034   00 EXEC     .COM 00
0035-0035   00 SYSGEN   .COM 00 : 0036-0036   00 FMAP     .COM 00
Group Allocation Number   Filename   Filetype   Extent Number   User Number

020F-020F   00 PBIOS    .ASM 03 : 0210-0216   01 ALIENS   .COM 00
0217-0218   00 PBIOS    .ASM 03 : 0219-0219   00 MOVCPM   .COM 00
021A-021A   00 MNTR     .DOC 00 : 021B-021C   00 PRNT     .DOC 00
021D-0220   00 REQ'D-SW.DOC 00 : 0221-0228   00 SW-DEF   .TXT 00
0229-0230   00 SW-DEF   .TXT 01 : 0231-0238   00 SW-DEF   .TXT 02
0239-023E   00 MOVCPM   .COM 00 : 023F-0241   00 FORMAT   .ASM 00
0242-0242   00 FORMAT   .ASM 01 : 0243-0247      ++FREE++
0248-024B   00 PBIOS    .ASM 04 : 024C-0253   00 SETATR   .ASM 00
0254-0254   00 PBIOS    .ASM 04 : 0255-0257      ++FREE++

G=0000:00,  T=4,  S=1,  PS=0
```

Figure 1. *This is a partial disk map for a CompuPro 8/16 system disk, showing the group allocation number, filename, extension ("filetype"), extent number and user number for each file.*



```
Group Allocation Number   Sector Block Number (always 128 bytes)
Physical Track   Logical Sector   Physical Sector

G=009C:01,  T=43,  S=2,  PS=1
G=009C:02,  T=43,  S=3,  PS=2
G=009C:03,  T=43,  S=4,  PS=3
G=009C:04,  T=43,  S=5,  PS=4
G=009C:05,  T=43,  S=6,  PS=5
G=009C:06,  T=43,  S=7,  PS=6
G=009C:07,  T=43,  S=8,  PS=7
G=009C:08,  T=43,  S=9,  PS=24
G=009C:09,  T=43,  S=10,  PS=25
G=009C:0A,  T=43,  S=11,  PS=26
G=009C:0B,  T=43,  S=12,  PS=27
G=009C:0C,  T=43,  S=13,  PS=28
G=009C:0D,  T=43,  S=14,  PS=29
G=009C:0E,  T=43,  S=15,  PS=30
G=009C:0F,  T=43,  S=16,  PS=31   End of 2K group number 009C.
G=009D:00,  T=43,  S=17,  PS=48   A new group allocation number starts
G=009D:01,  T=43,  S=18,  PS=49   after sector block number 16 (0FH)
(but may be different for your disk.)

31   End of 2K group number 009C
A new group allocation number starts
after sector block number 16 (0FH)
(but may be different for your disk.)
```

Figure 2. *Continuous incremental display of all group allocation, physical track, logical sector and physical sector assignments.*

indicators in this terse display:

G=009C:04, T=43, S=5, PS=4

G stands for group allocation and sector block number, T for physical track on the disk, S for logical sector number, and PS for physical sector number. To best illustrate what the display means, type the following command:

:+;/↵

— Advance to the next sector and repeat until you type a Control-*C*.

The group, track, logical sector, and physical sector assignments are displayed incrementally for your entire disk as shown in figure 2 (type Control-*C* to quit). "Sector blocks" are always 128 bytes each.

You move through the disk with Group (*Gnn*), Track (*Tnn*), and Sector (*Snn*) movement commands:

Gnn By allocation group. Example:

```
:G0↩
G=0000:00, T=2, S=1, PS=1
```

Tnn By track. Example:

```
:T8↩
G=0018:00, T=8, S=1, PS=1
```

Snn By sector. Example:

```
:S6↩
G=0018:05, T=8, S=6, PS=6
```

+nn Move ahead *nn* sectors. Example:

```
:+2↩
G=0018:07, T=8, S=8, PS=8
```

-nn Move back *nn* sectors. Example:

```
:-1↩
G=0018:06, T=8, S=7, PS=7
```

Input/Output

These commands let you read data from or write data directly to the disk, or to and from the buffer.

R Read a sector.
W Write a sector.
⟨ Save the current sector in a temporary buffer.
⟩ Recall the previously saved sector.

Displaying

G Show current group, track, sector.
M Map the entire disk (file group allocations). Another version:
Mnn Map starting at group *nn*.
D Display the current sector (display in hexadecimal and ASCII).
A Display the current sector in ASCII.
H Display the current sector in hexadecimal.
Vnn View *nn* sectors (as if you were using the CP/M TYPE command).
Show disk parameters. Example:

```
:#↩
Disk Information:
Tracks:            154
Sec/trk:           64
Grpsize:           16        (sectors per group)
Tot grps:          599
Dir entries:       256
Sys tracks:        4
```

Changing Disks and Data

CHnn,val
Change data in hexadecimal.
CAnn,val
Change data in ASCII (hexadecimal numbers enclosed within ⟨ and ⟩ symbols).
N Insert new disk.
Unn Change user directory number to *nn*.

Searching

Fname
Find a file in the directory.
F Find next occurrence (extent) of same name.
=aaaa
Scan for *aaaa* (ASCII characters) from current sector on.

Miscellaneous

Znn "Sleep" for *nn* tenths of a second to allow data viewing before it scrolls off.
Ld Log in disk *d* (drive letter A, B, etc.).
P Turn on/off printer output (echo console output).
Q Use before any command to make execution "quiet" (no console output).
X Exit to CP/M.
/nn Repeat previous command *nn* times (until you type ↑*C* if you omit the *nn*).

Quick Reference

Arguments in *italics* are optional; for numeric values, n or *n* is decimal, x or *x* is hexadecimal.

+n Move forward *n* sectors.
-n Move back *n* sectors.
Display disk parameters for the current drive.
=aaa
Search for the ASCII characters *aaa* from current sector. Upper and lower case letters are different. Use ⟨*xx*⟩ for hexadecimal number *xx*. For example, to find "IN 0" use:

```
=⟨DB⟩⟨0⟩↩
```

To find "(tab)H,0(CR)(LF)" use:

```
=⟨9⟩H,0⟨D⟩⟨A⟩↩
```

⟨ Save current sector in the buffer.
⟩ Restore saved sector.
? Display help message.
Aff,tt
ASCII display "dump" of current sector, or from addresses *ff* to *tt*.
CHaddr,byte,byte...
Change address to *byte* (hexadecimal number).
CAaddr,char...
Change address to *char* (ASCII character, ⟨*xx*⟩ allowed

	for hexadecimal numbers).
CHff-tt,byte	Change addresses from *ff* through *tt* to hexadecimal *byte* Example: *CH0-7F,E5*↩
CAff-tt,char	Change addresses from *ff* through *tt* to ASCII *char* Example: *CA0-7F,A*↩
Dff,tt	Display addresses in hexadecimal and ASCII.
Fname.ext	Find a file by its name.
Gnn	CP/M Allocation Group *nn*.
Hff,tt	Display in hexadecimal.
L	Log in disk in current drive.
Ld	Log in disk in drive *d*
Mnn	Map from group *nn* or entire disk.
N	New disk.
P	Printer echo on/off switch.
Q	Quiet mode (no console output).
R	Read current sector from disk.
Snn	Move to sector *nn*.
Tnn	Move to track *nn*.
Unn	Set User *nn* for Find command (CP/M 2.2 only).
Vnn	View *nn* sectors (display in ASCII).
W	Write the current sector to disk.
X	Exit DU program, return to CP/M.
Y	Yank current sector into sequential memory.
Znn	Sleep *nn* tenths of a second.
/nn	Repeat *nn* times (decimal).

Cancel a function with *C* or Control-*C* (↑*C*). Suspend output with *S* or Control-*S*. Separate DU commands on a multiple command line with a semicolon (;). Example:

:g0↩
:+;d;z#20;/↩
— moves in, displays, pauses for 2 seconds, and repeats the entire command until you type a Control-*C*. All numbers are in hexadecimal unless you precede them with *#*.

Command Summary

#	Display information about the disk (sectors per track, number of tracks, etc.).
+	Move forward one sector (if below track 2, this moves you to next numerical, if 2 or greater, this moves you based on CP/M's normal sector scrambling algorithm — + gets you to the next logical sector of the file.
-	Move backward one logical sector. You can include a decimal number with + or - to move forward or backward by several sectors. For example, *+15*↩ moves you forward by 15 logical sectors.
/nn	Repeat the entire command an optional *nn* times, or forever (until you type a Control-*C*). The number for *nn* can be from 2 to 65535.
⟨	Save the current sector in a save buffer, and reset the buffer pointer used by ⟨⟨ and ⟩⟩ commands.
⟨⟨	Save the current sector and increment the memory pointer so that the next ⟨⟨ command saves to the next buffer. Use the ⟨ command to reset, or the ⟩⟩ command to retrieve buffers sequentially, in order to move several sectors from one place on disk to another, or to another disk, or to memory (at 2000H where DDT can be used to access them).
=string	Search for ASCII-character *string*, starting at the current sector. You can use ⟨*xx*⟩ to represent a hexadecimal number xxH in the search *string*. In fact, DU ignores "bit number 7" (the "eighth bit," or parity bit) and treats all characters as ASCII unless you use ⟨*xx*⟩. Since *;* is a command delimiter, you have to use ⟨*3B*⟩ to search for a semicolon. Since ⟨ is a special character, use ⟨⟨ to represent a single ⟨ in *string*. Also note that the special symbol @ contains the displacement at which the match occurred. You can use it in a *C* command, as in: *:=LIX;CA* *,LXI;W*↩ — this searches for the string *LIX*, changes it to *LXI*, and writes the sector back to disk.
⟩	Get the saved buffer. The ⟨ and ⟩ commands can be used to move a sector to another place.
⟩⟩	Restore the "oldest" unrestored sector saved by ⟨⟨ command. This command may be placed in the middle of an infinite repeat */* command, since it will stop operating when there are no more sectors in the buffer.
?	Display the command summary.
A	Display ("dump") the current sector, ASCII characters only.
CHaddr,val,val,val...	Change to hexadecimal *val* in current sector starting at the *addr*.
CAaddr,string...	Change to ASCII *string* in current sector starting at the *addr*. Use ⟨*xx*⟩ for hexadecimal number xxH in *string* Use the *W* command to write changes to disk. The *C* command "echoes" for verification when you overlay existing data.

CHff-tt,byte or *CAff-tt,char*	Change a range of addresses, from address *ff* to *tt* (e.g., *05-E5*), to either a hexadecimal *byte* or an ASCII *char.*
D	Display ("dump") the current sector in hexadecimal and ASCII.
Fname.ext	Display the directory entry for the file named. You are subsequently moved to the file's directory sector.
F	Find the next occurrence of name in the directory (the next extent).
Gnn	Move to group *nn. and read.*
G	Show your current position.
H	Display ("dump") the current sector in hexadecimal only.
L	"Log-in" the current disk again. You may replace a disk with a new one, and use *L* to log-in the new disk.
Ld	"Log-in" disk *d* (as in *LB*↵ for drive B).
M	Display a map of the group allocations for files on the entire disk.
Mnn	Show which file is allocated to group *nn.*
N	Reset CP/M through its BDOS module. This may make it possible under some implementations of CP/M to change the disk format (i.e., disk density, one-sided to two-sided disks, etc).
P	Turn on or off the printer "echo" of console input and output.
Q	"Quiet" mode. Preceeding any command, this command suppresses console output.
R	Reads the current sector (currently positioned to) into memory. *R* (read) is implicit in the *G*, +, and - commands, but *not* in the *S* and *T* commands.
Snn	Move to to sector *nn* and read.
Tnn	Move ("seek") to track *nn* (no read).
Un	"Log-in" user *n* for the next *F* command. This command displays the ? error if you are not using CP/M-80 version 2.0 (or greater).
V	View the current sector (assumes ASCII data and displays like the CP/M TYPE command).
Vnn	View *nn* sectors.
W	Write the current sector back to disk. Note: this command should not be used after using an *F* command, since *F* uses CP/M to find the file in the directory.
X	Exit DU, return to CP/M.
Znn	Sleep — cause the program to pause, to let you look at a display before it scrolls past. *Z* is one second. *Znn* sets the pause to *nn* tenths of a second on a 2 MHz 8080 CPU.

Notes on Using DU

1. Erasing a Disk Directory

The following commands will erase all files on drive B by changing the B: disk directory to E5H ("H" stands for hexadecimal):

:LB↵	("Log-in" the B drive)
:G0↵	(Move to start of directory)
:CH0-7F,E5↵	(Fill with E5H)
:(↵	(Save the sector)
:);W;+;/16↵	(Restore, write, move to next and repeat 16 times)

This example could be shortened to:

:LB;G0;CH0-7F,E5;(↵
:);W;+;/16↵

2. Display Commands

You can add to a display command (*D*, *A* and *H*) an optional starting and ending address:

:D0,7F↵	(This is the same as *D*↵)
:D3,5↵	
:A20,3F↵	

3. Logging In a Disk

Some problems may occur when "logging in" (setting as the current disk) a mixed-density disk (e.g., track 00 is single density, and the directory tracks are double density, as in IBM System 34 disks).

One solution is to first "log-in" a disk that you know will work, and that is also of the same density as the disk that you want to examine and alter. Then put in the new disk *without* performing a "log-in." However, if you do this, you are opening yourself up to possible problems related to the buffering of physical sectors in the BIOS.

Another technique (not guaranteed) is to seek to the unused inner tracks of the first disk, then do the read, *and then*

WIZARD of OsZ™
a division of Contact Enterprises™

New Releases in Stock and Immediately Available

'Software and Accessories for Osborne™ Computers is our *only* business'

CHEKWIZ™ — the ideal financial tool for home or office
Handles over 1400 check entries and allows quick search by category, date, or name. Search and itemize a year's checks; ideal at tax time. Handles amounts up to 99,999,999.99. Allows easy entry of service charges, interest, automatic withdrawals, and provides automatic backup to protect against inadvertant loss of data............List $50

PLINK™ & RCPMLINK™ — now on one disk
Transmit complete file to bulletin board in seconds with user friendly PLINK. Saves files to buffer and writes to drive A or B. RCPMLINK enables fast, accurate block transfer of files between remote CP/M systems (or other Osborne 1 Computers)...................List $50

SIGNALMAN™ MARK V MODEM — Source℠ subscription included
A small, light, compact direct connect modem for Osborne 1™. Just plug it in and communicate through any modular telephone. Complete with cables and 9-pin Osborne wired connector. Uses 9v internal battery; 9v dc adapter is available...............................List $129

GAMES DISK 4 — graphics at play
2DMAZE, 3DMAZE, BOUNCE and LANDER (by Mirashanta). Programmed in CBASIC. Games are in two levels, simple then advanced. All include extensive graphics...............................List $50

GAMES DISK 5 — from Casino to Deep Space
Play Black Jack against the computer, the house limit is $10,000. Or how about a return to the old west to try your hand as a miner? Goldmine has all the rewards and risks you can handle. Then jump into the future with an exciting game of Space War. For a real challenge, try to win at Wizdot. Match wits with the computer, add line segments to make squares. Make 5 and you win....................................List $50

GAMES DISK 6 — educational fun from pre-school to college
Ghost in the Chimney delights pre-schoolers with visible rewards for right answers. Dingo the Cookie Goblin teaches addition and subtraction in a subtle form. Supermind pits your skills against the computer which hides 5 random letters; you have 13 tries to guess the letters in order....List $50

Write or Call WIZARD OF OsZ™ Your Source for Osborne™ Software
Add $2.50 shipping & handling   Dealer Inquiries Invited   Calif. add 6½ % sales tax
9614C Cozycroft Ave.   Chatsworth, California 91311   Telephone: (213) 709-6969

Circle reader service number 13

change disks. If anything is written, you will not have destroyed anything — assuming the disk is not completely full.

Another possible solution, assuming the second disk does not contain a CP/M system, would be to seek to track 1, do the read operation there, and then change disks to the new one.

4. Sector Buffers

You can store up to 255 sectors in a buffer with the ⟨⟨ command; use ⟩⟩ for every ⟨⟨ to make the buffer contents available for writing. To reset (throw away the contents of) the buffer, use one ⟨ command.

The *M* command (directory map) also uses an area of memory for a buffer. To minimize problems, the map buffer is placed at the end of the currently highest used ⟨⟨ buffer.

In small systems, where many buffers have been saved using ⟨⟨, the *M* command might report that it ran out of memory. By executing the ⟨ or ⟩⟩ command a sufficient number of times for it to tell you there are no more sectors in the buffer, you can make room available in memory at a lower address, and *M* can again be used for "mapping" the directory.

Interpreting CP/M Directory Data

Single Density

The following explains the format of a CP/M directory entry, as shown by DU using either the *F* (find file) command or using *D* to display ("dump") the directory sectors, which are located in groups 0 and 1 on a single-G density disk.

Figure 3 shows the result of the command *FSID.COM*⏎ command (find the SID.COM file).

A file *extent* indicates to CP/M the number of 16 thousand (actually 16K) consecutive bytes contained within a file. Extents are numbered from 0 to 31. One extent may con-

```
40   00534944 20202020  20434F4D 0000003A  *.SID     COM...:*
50   33343536 3738393A  00000000 00000000  *3456789:........*
```

First line

```
40   00534944 20202020  20434F4D 0000003A  *.SID     COM...:*
```

Filename in hexadecimal   Filename in ASCII   Extent   File Size in Sectors
00 = File Active (other values represent user area number; e.g., 03 = user area number 3).
E5 = File Erased
Displacement of the "line" (16 entries/line) in the directory sector

Second line

```
50   33343536 3738393A  00000000 00000000  *3456789:........*
```

Allocation groups that happened to be printable.
Allocation group numbers

Figure 3. *Using the find command to find the file SID.COM on a single-density disk.*

```
:FSID.COM↩                                                    (Find SID.COM on the disk.)

00  00534944 20202020  20434F4D 0000003A  *.SID     COM...:*
10  38003900 3A003B00  00000000 00000000  *8.9.:.;.........*
G=0000:00, T=2, S=1, PS=0
```

Figure 4. *Using the find command on a double-density disk.*

tain one, two, four, eight or sixteen blocks. A *block* is the basic unit of disk space allocation, and may be 1K, 2K, 4K, 8K or 16K consecutive bytes.

These blocks are assigned *group numbers* (a "group" is a set of disk sectors). Group numbers are allocated dynamically to accommodate the required disk space used by a given file (group 0 is *always* the start of the directory group).

More than one filename might appear in the directory for one file, depending on the number of extents required to accommodate the file's size. (See figure 1 for an example of this — PBIOS.ASM has two entries in the directory for one file.)

Double Density

Figure 4 shows an example of finding SID.COM running on a double-density disk system.

The major difference between single and double density is that the groups now occupy 2 bytes. This convention follows the Intel and CP/M convention of putting 16 bit values with the high byte first: "38 00", "39 00", etc. means groups 0038, 0039, etc.

Note also that in double-density disks, each group represents 2K, not 1K, so there are half as many groups for the same file.

Be *very careful* when patching a double-density disk directory. I once made the mistake of typing the following command:

```
:CH10,38,39,3A,3B... ↩
```

When I tried to read this file, DU tried to access group 3938, causing angry exclamations from the disk stepper as it attempted to go south to Peoria for the data.

DU Examples

Ed. note: the following dialogue was recorded as close to word for word as our editorial standards would allow. It is important to get the "feel" of this program. Some say this program is not for beginners; but why then would anybody be using it? We do promise a DU tutorial, as soon as we've learned more about using it.

1. Recovery of an Erased File

Suppose you have just added the final touches to your latest and greatest program or text file. Being of usual slothful weekend mind and spirit, you (of course) have *never* made a backup copy after having invested thousands of weekend hours into it. You proudly examine the directory of your disk:

```
A)DIR B:↩
B:BIZ-INIT   BAS : BIZ-CALC   BAS
B:BIZ-CHRT   BAS : BIZ-WIZ    TXT
B:BIZ-WIZ    BAS
```

Yep... all there, and... (ring, ring... its the phone again!). OK, I'll be back in a few minutes and print the file for posterity... (tick, tick... minutes turn to hours...).

Ah yes, let's see where was I? Oh yes, that guy called and wanted to modem over his latest version of FOOBAR... Fine, better make some room on this "scratch disk" in the B: drive...

So you do this abominable command:

```
A)ERA B:*.*↩
```

— (Get rid of whatever junk is there.)

```
ALL (Y/N)?Y↩
```

— (Sure, can't remember what's there, can't be important.)

```
A)DIR B:↩
```

— (and just to be sure...)

```
NO FILE
```

— (Yep! No files left... but... AAaarrgghh! The wrong disk!!!)

What do you do?

DU, the disk utility to the rescue!

I just *knew* something clever would eventually emerge from my beer-fogged brain to save my ASCII... So here we go:

```
A0)DU↩

DISK UTILITY ver 7.7
Universal Version

Type ? for help
Type X to exit

:LB↩
```

— ("Log-in" disk B — a forest reclamation project!)

```
:G0;D↩
```

— (Go to Group 0, always the directory group, and dump:)

```
G=00:00, T=2, S=1, PS=1
00   E542495A 2D494E49  54424153 0000001C  *eBIZ-INITBAS....*
10   02030405 00000000  00000000 00000000  *................*
20   E542495A 2D43414C  43424153 00000050  *eBIZ-CALCBAS...P*
30   06070809 0A0B0C0D  0E0F0000 00000000  *................*
40   E542495A 2D434852  54424153 00000019  *eBIZ-CHRTBAS....*
50   10111213 00000000  00000000 00000000  *................*
60   E542495A 2D57495A  20545854 00000029  *eBIZ-WIZ TXT...)*
70   14151617 18190000  00000000 00000000  *................*
```

Hmmm... Those are my files, yep, the ones with funny little e 's in front of them... Let's also look at the next directory sector:

:+;D↩

— (Move +1 sector and Dump it also:)

```
G=00:01, T=2, S=2, PS=7
00   E542495A 2D57495A  20424153 00000080  *eBIZ-WIZ BAS....*
10   1A1B1C1D 1E1F2021  22232425 26272829  *...... !"#$%&'()*
20   E542495A 2D57495A  20424153 0100006B  *eBIZ-WIZ BAS...k*
30   2A2B2C2D 2E2F3031  32333435 36370000  **+,-./01234567..*
40   E5E5E5E5 E5E5E5E5  E5E5E5E5 E5E5E5E5  *eeeeeeeeeeeeeeee*
50   E5E5E5E5 E5E5E5E5  E5E5E5E5 E5E5E5E5  *eeeeeeeeeeeeeeee*
60   E5E5E5E5 E5E5E5E5  E5E5E5E5 E5E5E5E5  *eeeeeeeeeeeeeeee*
70   E5E5E5E5 E5E5E5E5  E5E5E5E5 E5E5E5E5  *eeeeeeeeeeeeeeee*
```

Well, it's all sitting there in the directory... Let's move back one sector, and start fixing things to recover those erased files:

:-↩ (Move back one sector.)

```
G=00:00, T=2, S=1, PS=1
```

:*CH0,0;CH20,0;CH40,0;CH60,0*↩

— (Change all `E5` 's to zeros:)

E5E5E5E5

— (The `E5` 's (erased file flags) evaporate before your very eyes!)

:*D*↩

— (Dump it just to check...)

```
00 0042495A 2D494E49 54424153 0000001C *.BIZ-INITBAS....*
10 02030405 00000000 00000000 00000000 *................*
20 0042495A 2D43414C 43424153 00000050 *.BIZ-CALCBAS...P*
30 06070809 0A0B0C0D 0E0F0000 00000000 *................*
40 0042495A 2D434852 54424153 00000019 *.BIZ-CHRTBAS....*
50 10111213 00000000 00000000 00000000 *................*
60 0042495A 2D57495A 20545854 00000029 *.BIZ-WIZ TXT...)*
70 14151617 18190000 00000000 00000000 *................*
```

```
:W↵
```

— (Write this directory sector back to the disk:)

```
:+↵
```

— (Move to the next directory sector:)

```
G=00:01, T=2, S=2, PS=7
```

```
:D↵
```

(Dump it also, to be sure...)

```
00   E542495A 2D57495A  20424153 00000080  *eBIZ-WIZ BAS....*
10   1A1B1C1D 1E1F2021  22232425 26272829  *...... !''#$%&'()*
20   E542495A 2D57495A  20424153 0100006B  *eBIZ-WIZ BAS...k*
30   2A2B2C2D 2E2F3031  32333435 36370000  **+,-./01234567..*
40   E5E5E5E5 E5E5E5E5  E5E5E5E5 E5E5E5E5  *eeeeeeeeeeeeeeee*
50   E5E5E5E5 E5E5E5E5  E5E5E5E5 E5E5E5E5  *eeeeeeeeeeeeeeee*
60   E5E5E5E5 E5E5E5E5  E5E5E5E5 E5E5E5E5  *eeeeeeeeeeeeeeee*
70   E5E5E5E5 E5E5E5E5  E5E5E5E5 E5E5E5E5  *eeeeeeeeeeeeeeee*
```

```
:CH0,0;CH20,0↵
```

— (Change the two directory entries...)

```
E5E5
```

```
:D↵
```

(To be sure!)

```
00   0042495A 2D57495A  20424153 00000080  *.BIZ-WIZ BAS....*
10   1A1B1C1D 1E1F2021  22232425 26272829  *...... !''#$%&'()*
20   0042495A 2D57495A  20424153 0100006B  *.BIZ-WIZ BAS...k*
30   2A2B2C2D 2E2F3031  32333435 36370000  **+,-./01234567..*
40   E5E5E5E5 E5E5E5E5  E5E5E5E5 E5E5E5E5  *eeeeeeeeeeeeeeee*
50   E5E5E5E5 E5E5E5E5  E5E5E5E5 E5E5E5E5  *eeeeeeeeeeeeeeee*
60   E5E5E5E5 E5E5E5E5  E5E5E5E5 E5E5E5E5  *eeeeeeeeeeeeeeee*
70   E5E5E5E5 E5E5E5E5  E5E5E5E5 E5E5E5E5  *eeeeeeeeeeeeeeee*
```

```
:W↵
```

— (Write this directory sector as well, before we quit!)

```
:X↵
```

— (Exit to CP/M-80...

```
A>DIR B:↵
```

— (As if by magic...)

```
B:BIZ-INIT   BAS : BIZ-CALC  BAS
B:BIZ-CHRT   BAS : BIZ-WIZ   TXT
B:BIZ-WIZ    BAS

A>
```

...it's all done with mirrors!

2. Modifying a Disk For Program Autoload

By modifying a portion of the CP/M operating system, you can force it to automatically load and execute a command file every time CP/M starts up from a cold start or reset.

This feature can be particularly useful if you want to set up some fancy printer initialization, modify a serial port baud rate or data format parameters, or start a menu program after every cold start or reset.

To enter an "autoload" filename to a CP/M disk (some of the data in this example is representative of my disk only):

```
:T2;S1;D↵
```

— (Move to track 2, sector 1, and dump...)

```
 T=2, S=1, PS=0
 T=2, S=1, PS=0
00  C35CDDC3 58DD7F00  20202020 20202020  *C\]CX]..        *
10  20202020 20202020  434F5059 52494748  *         COPYRIGH*
20  54202843 29203139  37392C20 44494749  *T (C) 1979, DIGI*
30  54414C20 52455345  41524348 20200000  *TAL RESEARCH  ..*
40  00000000 00000000  00000000 00000000  *................*
50  00000000 00000000  00000000 00000000  *................*
60  00000000 00000000  00000000 00000000  *................*
70  00000000 00000000  00000000 00000000  *................*
```

:CA8,SIGNON⟨0⟩ ↩
— (Change ASCII, address 8 to SIGNON and 0.)

:CH7,6 ↩
— (Change HEX, address 7 to a value of 6 — the length byte.)
00
— (What the value was prior to the change.)

:D ↩
— (Dump the disk sector for display:)

```
00  C35CDDC3 58DD7F06  5349474E 4F4E0020  *C\]CX]..SIGNON. *
10  20202020 20202020  434F5059 52494748  *         COPYRIGH*
20  54202843 29203139  37392C20 44494749  *T (C) 1979, DIGI*
30  54414C20 52455345  41524348 20200000  *TAL RESEARCH  ..*
40  00000000 00000000  00000000 00000000  *................*
50  00000000 00000000  00000000 00000000  *................*
60  00000000 00000000  00000000 00000000  *................*
70  00000000 00000000  00000000 00000000  *................*
```

:W ↩ (Write the disk sector.)
:X ↩ (Exit to CP/M.)

You can now reset or "cold start" this CP/M system disk, and it will "autoload" and run the SIGNON program prior to doing anything else. Of course, substitute your own program name for SIGNON in the above example.

3. A "Poor Man's" Disk Test

DU can also function as a "Poor Man's" disk test that can perform simple sequential write/read/seek functions with user-specified data patterns. Here, in very simple form, is a four-sector non-destructive read-only/seek test. Any subsequent errors would be trapped by the CP/M BDOS and stop the test with whatever error message was appropriate (e.g., BDOS Err on B: BAD SECTOR):

:T20;S1 ↩

```
G=0040:00, T=20, S=1, PS=0
```

:+;+;+;+;-;-;-;-;/ ↩
— (Read forward four sectors, read backward four sectors, and repeat (until you type a ↑*C*).

```
G=0040:01, T=20, S=2, PS=1
G=0040:02, T=20, S=3, PS=2
G=0040:03, T=20, S=4, PS=3
G=0040:04, T=20, S=5, PS=4
G=0040:03, T=20, S=4, PS=3
G=0040:02, T=20, S=3, PS=2
G=0040:01, T=20, S=2, PS=1
G=0040:00, T=20, S=1, PS=0
G=0040:01, T=20, S=2, PS=1
G=0040:02, T=20, S=3, PS=2
G=0040:03, T=20, S=4, PS=3
G=0040:04, T=20, S=5, PS=4
G=0040:03, T=20, S=4, PS=3
```

Days or weeks later...

```
G=0040:03, T=20, S=4, PS=3
G=0040:02, T=20, S=3, PS=2
G=0040:01, T=20, S=2, PS=1
G=0040:00, T=20, S=1, PS=0
G=0040:01,
```

— (Interrupted with ↑*C* from the keyboard.)

By using your imagination, you can come up with a variety of useful applications for DU.

Floppy Disks: What They're Made Of, And How To Care For Them

"Are these floppies, flippies, discs or diskettes?"
Floppy Disks Are Unveiled and Explained

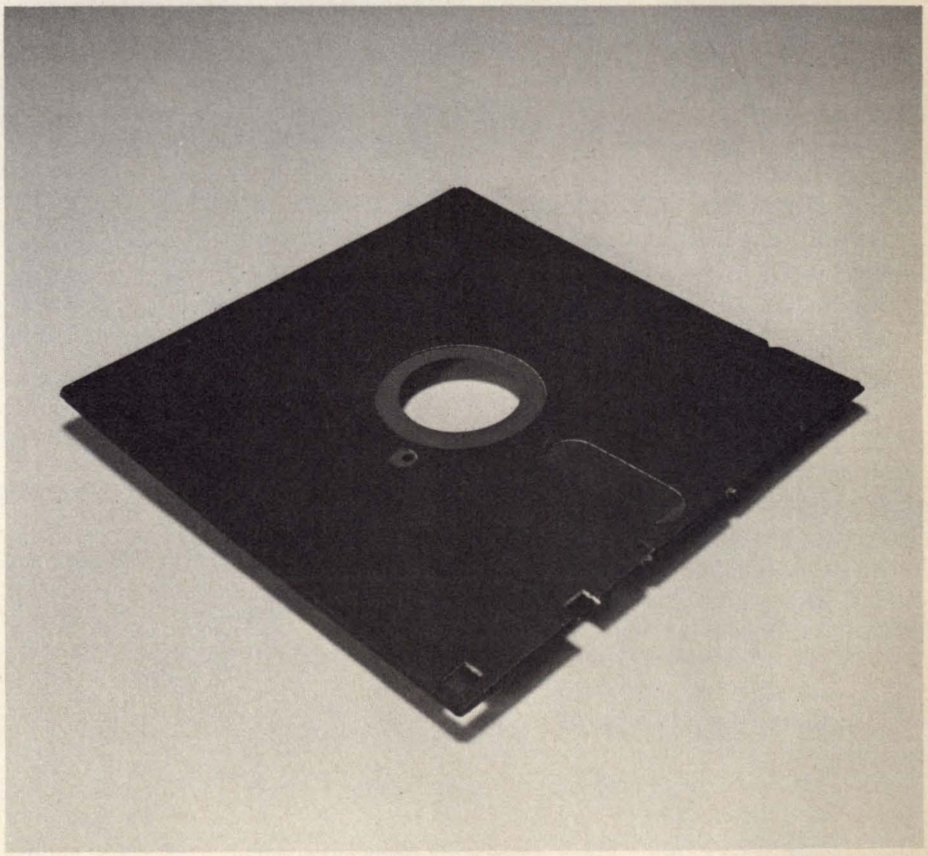
by Steve Rosenthal
illustrations by Phi Bach Pham

The floppy disk is what made CP/M possible, but in some ways it also made CP/M necessary. CP/M was one of the first, and certainly the most successful, *floppy disk* operating system.

It is a program that loads itself from a floppy disk and controls the computer — lets you run your programs, lets you name your files, etc. It "thinks" of memory storage in terms of space on floppy disks.

A *floppy disk* can store hundreds of thousands of characters. It needs precise, detailed positioning information to find any particular piece of data. When a computer manufacturer installs a version of CP/M in a computer, the manufacturer modifies CP/M to have this detailed positioning information. From that point on, CP/M can manage things to present to you a list of files with names you can read, and methods to access the data on the floppies.

Let's take a closer look at these floppy disks.



Introducing the floppy

A floppy disk (figure 1) is a thin circular sheet of plastic coated with magnetic material, all contained in a square plastic jacket — a cross between a phonograph record and an audio cassette tape. The disk stays in the jacket, which has one slot cut out for access to the magnetic surface plus a few small but crucial holes for sensing the position of the disk.

Some people call floppy disks "diskettes" to differentiate them from large, rigid disks. But "floppy," "floppy disk," "flexible disk," "disk" and "disc" are all in common use.

A floppy is placed in a disk drive (figure 2), the recorder-playback machine which does the actual reading and writing. Both disks and drives come in two standard sizes (5.25-inches across and eight inches across), with a number of smaller sizes now being introduced. Like 78 rpm records and 33 ⅓ LPs, the two sizes run at different speeds, use slightly different formats, and can hold different amounts of material. Unfortunately, drives are not made like record players, and you can't slip on a 5-inch adapter and play your small disks on a big drive.

The disks are somewhat flexible, but sharp bending will put creases in the jacket and make them unusable. Keep disks out of dust, direct sunlight (because they heat up), extreme temperatures, and strong magnetic fields. Apparently, you can safely lay a disk (preferably in its dust-jacket outer envelope) on top of a disk drive, but data can get scrambled if the disk is left on large motors or power supplies, or near magnets such as the one in your telephone.

Data is stored on floppy disks in a series of concentric circles, with the outer circle known as track 0 and the inner dependent on the track spacing and size of the disk (figure 3). Most eight-inch disks use 77 tracks per disk, while common capacities for five-inch are 35, 40 or 80 tracks.

On each track, the data is further broken up into "sectors," which are arcs (lengths along the circumference) along each track. Most disks for small computers store a multiple of 128 bytes in each sector (such as 128, 256, 512 or 1024).

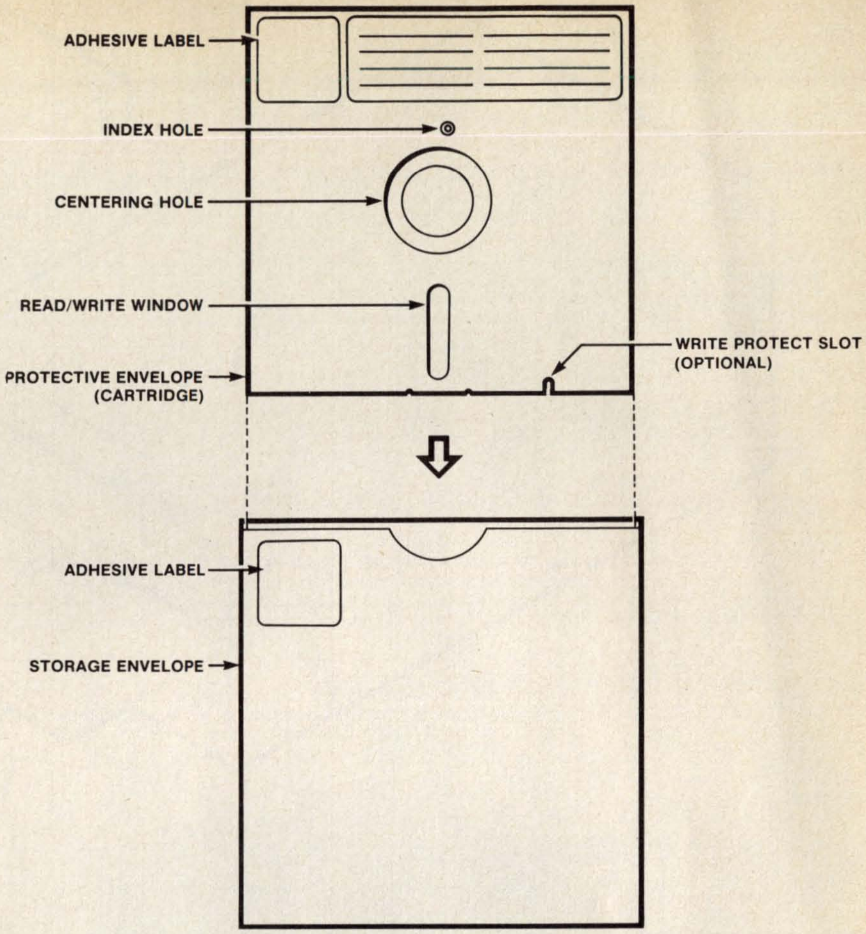


Figure 1. *A floppy disk is a round flat piece of recording material permanently sealed in a square protective jacket. When not in use, a floppy disk should be placed in its protective paper or plastic envelope. Data is written on or read from the disk by a floppy disk drive.*

The computer reads or writes one sector at a time. If it wants smaller amounts of data, it reads a whole sector and discards the remainder. If the computer wants to change only part of a sector, it must first read, then update, and then send the updated sector back to the disk.

The first crop of disk drives wrote and read only on the bottom surface of the disk (the side away from the label). The most recent generation of disk drives use both sides. Disks are manufactured for compatibility with two-sided drives, although you pay a premium for ones which are guaranteed to also be free from magnetic flaws on the second side.

Disks also come in "flippy" models — meaning that to access the second side you remove the disk from the drive and turn it over to present the other side to the read/write head.

Disk Formats

Besides size, there are five additional important distinctions which characterize disks and drives. These are:

- How reference points, which are needed to find the position of any given piece of data, are located for the drive to lock into position;
- How data is arranged on the disk (type of *sectoring*);
- How dense is data stored on disk (disk *density*);
- How are the binary 1's and 0's encoded into magnetic changes (disk *format*);
- Whether this drive writes on one or both sides of the disk.

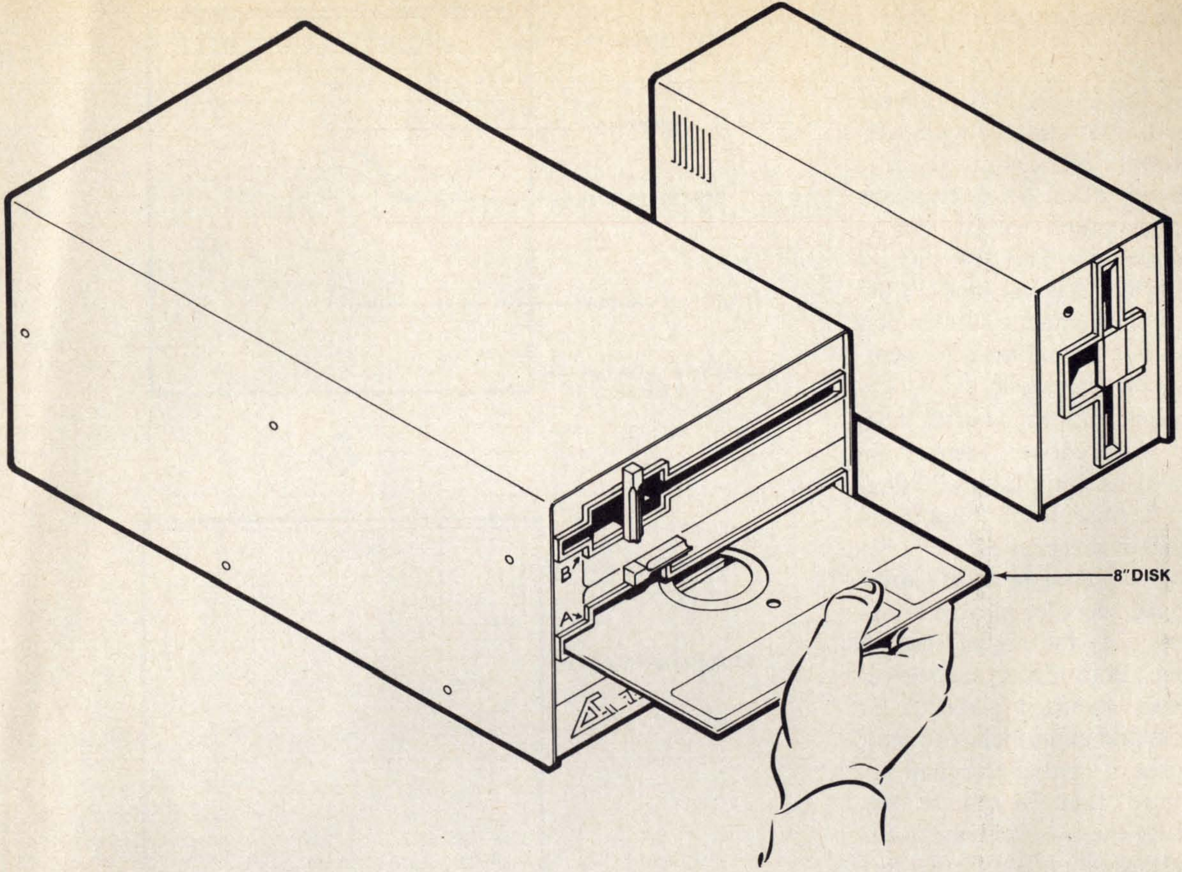


Figure 2. *Although there are more than a dozen different manufacturers of disk drives, most are similar in operation, appearance, and mounting. Drives, like disks, come in 5.25-inch and 8-inch models.*

Sectoring

The question of location is referred to as *sectoring*, and disk drives use either "soft" or "hard" sectoring. Soft sectoring, as shown in figure 4, means that one hole cut into the disk near the center is used to give a master reference point, and the rest of the way around the circle the positioning information is encoded into the material written magnetically on the disk.

Hard-sectored disks (figure 4) use a few dozen holes around the inner or outer disk rim, and are used only with hard-sector drives.

CP/M was written for soft-sectored disks, and almost all new systems using CP/M are of that type. Although hard-sectored disks can store more information (they don't use up data space with positioning information), they are less popular. However, CP/M has been made to run on hard-sectored machines as well.

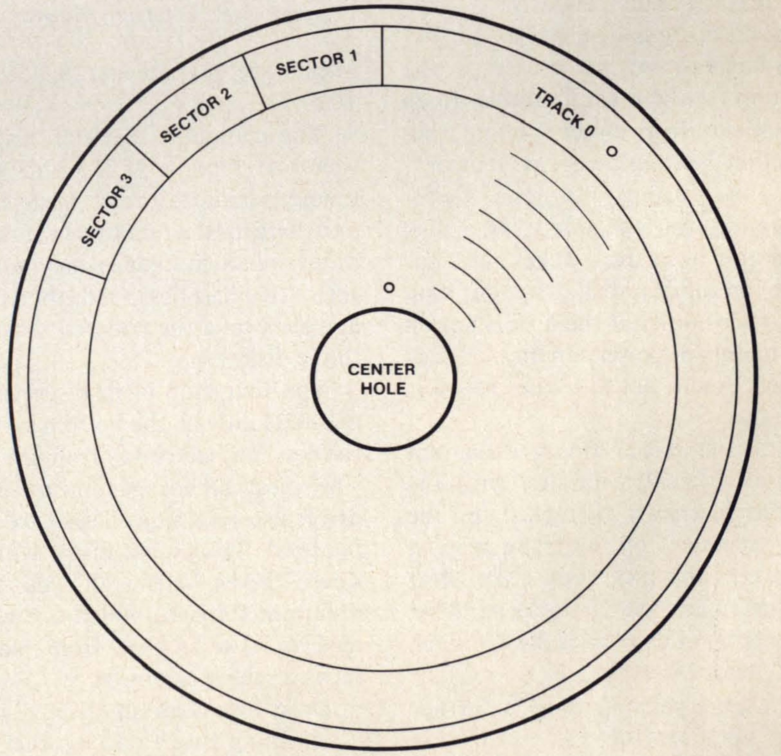


Figure 3. *Data is written on a disk in a series of concentric tracks. Track 0 starts at the outer edge, with the highest numbered track nearest the center. On each track, data is written in blocks called sectors.*

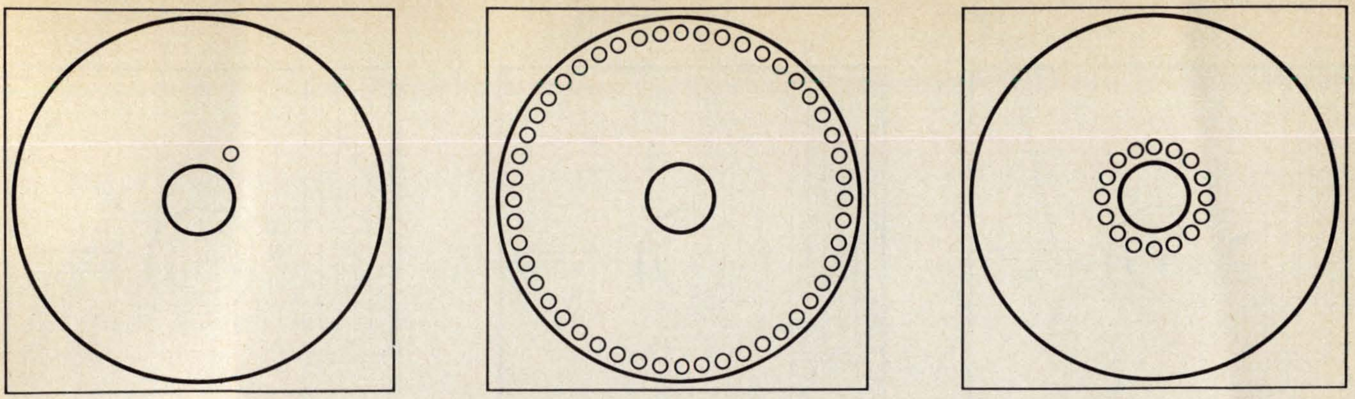


Figure 4. *Hard or soft sectors mark the blocks of data. With soft sectors, a single hole tells the drive where to start the track, and the rest of the positioning information is written magnetically on the disk. With hard sectors there are as many as several dozen holes either around the outside or inside of the disk to mark the start of each block.*

Formatting

Formatting refers to the way the data is arranged on the disk. Included is the number of tracks, number of sectors per track, and how many characters (*bytes*) are written in each sector.

It is largely due to different formats that computers cannot accept and use disks from other computers. A disk must have a format that is *compatible* with the computer's regular disks, or the computer must be able to read other disk formats besides its own.

Chances are that if you own an eight-inch disk CP/M system, you can read data from any "single-density CP/M standard" eight-inch disk formatted for another system.

This eight-inch "standard" is a format popularized by IBM for the 3740 disk drive, and is also known as "3740-compatible." Almost all eight-inch CP/M systems can be set to produce or read disks of this type.

However, IBM set a de-facto standard too early in the development of floppy disk drives. It soon became more practical to double the amount of data per unit area on the disk, rather than adhere to a single-density standard. Many double-density systems use unique proprietary formats, which means that double-density disks cannot be exchanged among such systems. The different formats account for the differing amounts of data that can be stored on various double-density disks.

Fortunately, drive and software technology provides ways for eight-inch systems to read and write 3740-compatible format disks as well as proprietary double-density formats.

Also, in the last few years many manufacturers of eight-inch systems have adopted something close to the IBM System 34 format. Disks from one computer of this type can often be used with other computers of this type.

In the 5-inch size, there is still more variety, but many companies are now offering systems which can be set for a number of formats, including the one used by the IBM Personal Computer. If two systems have at least one common format, they can easily exchange data (but this only means they can transmit data files to each other — it does not necessarily mean they can run the same programs).

To use a disk with any system, it must first be formatted for that system. Most blank disks must be correctly formatted before they can be used. Although it is possible to buy disks preformatted in the more popular arrangements, most systems include a program that will do this task given the correct, soft-sectored disk of the right size.

Density

Density refers to the proximity of sectors of data — how densely packed the disk is with data. There once was only one density, then someone invented a disk that was "double the density" of the first, and both types of disks were so named.

The "CP/M standard" single-density disk holds about 250,000 characters on each side of an eight-inch disk. The various double-density formats hold anywhere from 400,000 to 700,000 characters on each side.

Five-inch single-density disks hold about 100,000 characters on a side. Five-inch double-density formats hold anywhere from 200,000 to 600,000 characters per side.

As for the encoding, the exact differences are a bit arcane for most computer users. Almost all single-density disks use a method called frequency modulation (abbreviated FM), while most double-density disk use modified frequency modulation (MFM).

As users, the choice of methods is not apparent to us, and therefore not really a matter of concern. If disk interchange between computers is important for your computer application, you should conduct an actual test of this interchange, not rely on specifications or manufacturers' claims.

Two-Sided and Flippy Disks

Disks are made with recording surfaces on both sides, and then tested to see if they can store information on both sides. If both sides work, the disk

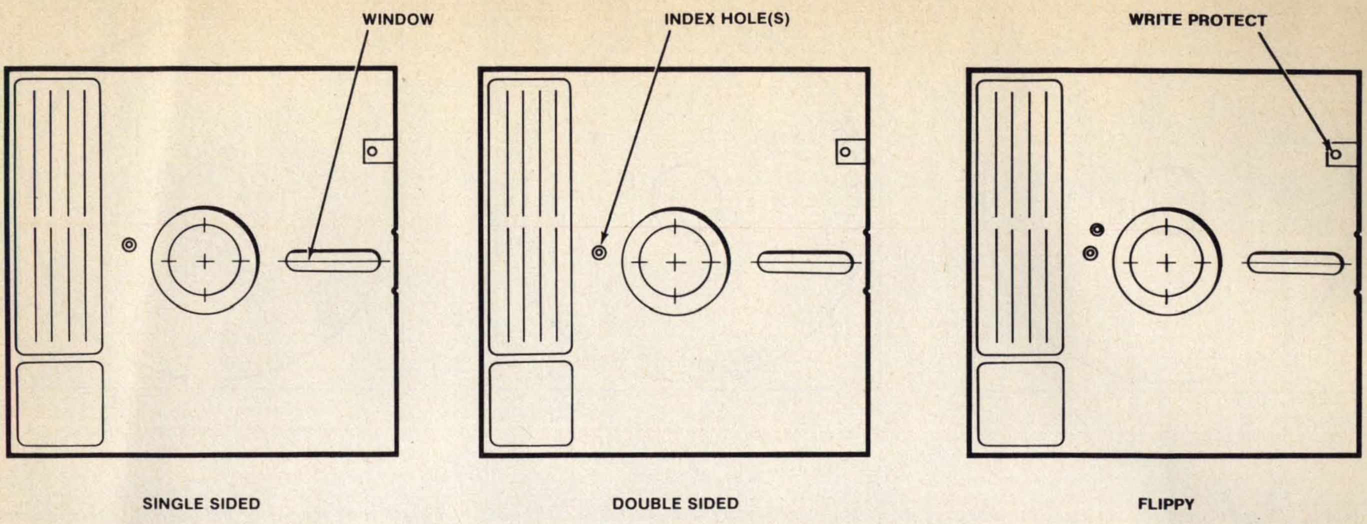


Figure 5. *One-sided and two-sided disks, with a "flippy" disk (can be used as both one-sided and two-sided, depending on the disk drive).*

manufacturer guarantees the disk for two-sided operation. If the label side does not work, the disk is only guaranteed for one-sided operation. If the side without the label does not work, the disk is rejected.

The difference between one-sided and two-sided disks, besides the guarantee, is where the small hole in the jacket is punched. This hole allows the drive to see an "index" hole marking the start of each track.

On one-sided disks, the index hole is nearly centered over the large hole in the middle of the disk, whereas the index hole on two-sided disks is more to the side (see figure 5). A two-sided disk can be turned into a one-sided or flippy disk by carefully punching new holes at the required positions, using a standard one-sided disk as a guide. Be careful, however, to punch only through the jacket, and not through the disk material.

Write Protection

Like recording tape, disks can be erased or written on at any time. However, the computer can order the drive to erase or re-write large amounts of information in a small fraction of a second (and before we could do anything to stop the process). Therefore, we need a way to make sure that this won't accidentally happen to vital information even if something goes wrong. The usual solution is to allow disks to be *write-protected*.

Eight-inch disks are write protected by uncovering an existing notch on the side farthest from the label, closest to the read/write window. If your disk does not have such a notch, you can cut one into the side of the disk, using the notch dimensions shown in figure 6.

This notch allows light from a LED inside the disk drive to shine through to a photocell. When this happens, a circuit inside the drive tells the rest of the circuitry to refuse write commands from the computer. A signal line also tells the computer the disk is write pro-

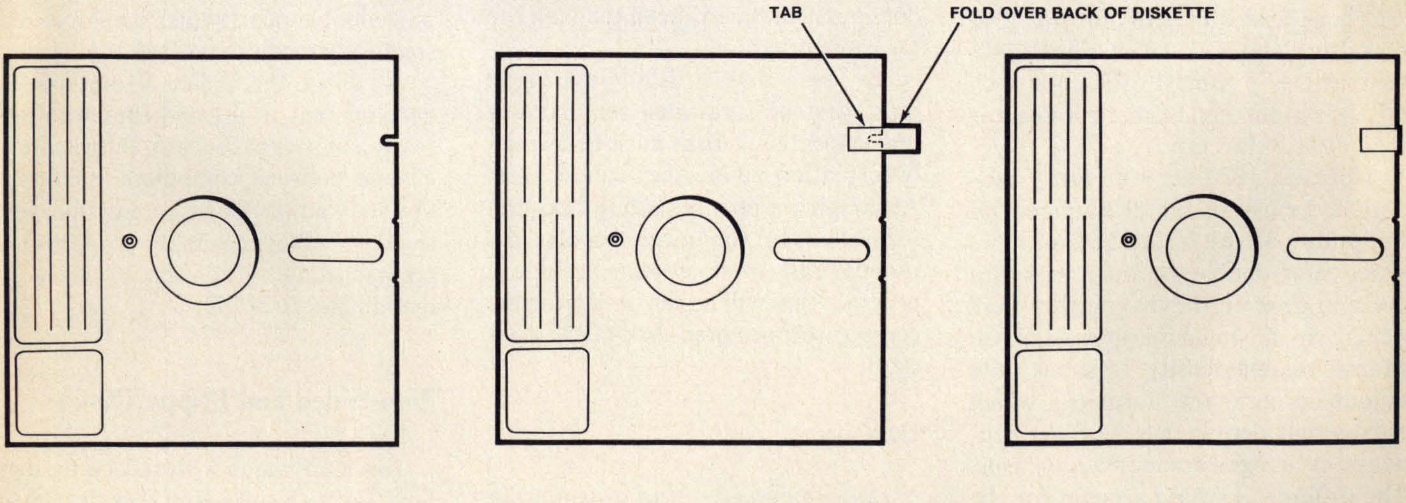


Figure 6. *Eight-inch disks are write-protected by cutting a notch in the end that is placed into the drive. Covering that notch allows the disk to be written upon. You fold a pressure-sensitive opaque tab over the notch of a new disk when you first format it for use. To alter data on the disk, a tab must be covering the notch.*

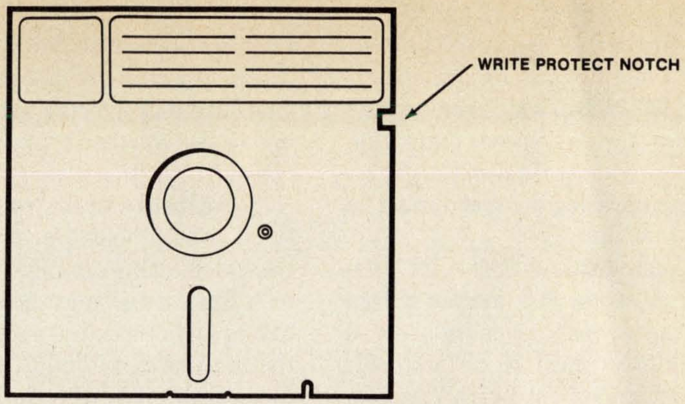
tected, but in standard CP/M systems this information is not passed through to the operating system or the user.

To allow the disk to be written on, this notch must be re-covered, usually with a pressure-sensitive opaque label. Although ordinary tape can be used, some colors are not sufficiently opaque, and many tapes such as electrical tape leave a sticky residue which may get into the disk jacket or drive.

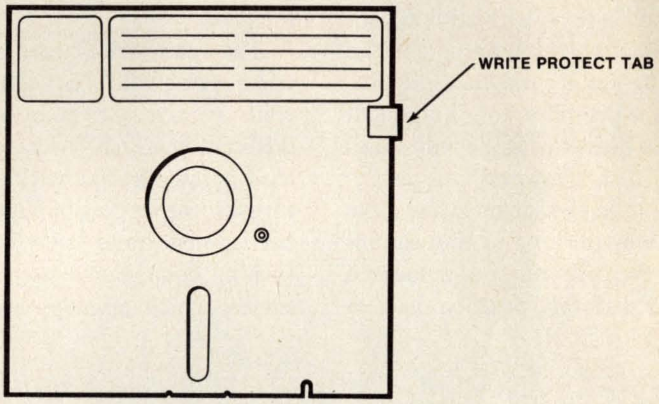
Write-protect notches can be punched into disks which do not have them with a standard 1/4 inch paper punch, using two overlapping holes. Fancy punches which create oval shaped notches are also available from stationery stores (the punches are used to punch bus transfers and tickets), but are several times as expensive.

On five-inch drives, the principle is the same but the details are reversed. The notch is on the side of the disk closer to the label, and it must be *covered* to be write-protected (see figure 7). Thus, an un-covered notch on a five-inch disk allows writing to the disk, while a covered notch prevents it (the opposite is true of eight-inch disks).

Why are the details just the opposite between five-inch and eight-inch disks? We don't know, but the inconsistency is typical of computer technology in the absence of standards.


WRITE PROTECT NOTCH

UNPROTECTED


WRITE PROTECT TAB

WRITE PROTECTED

Figure 7. *Five-inch disks are write-protected by covering a notch in the side of the disk. Uncovering the notch allows the disk to be written upon. This procedure is the opposite of the one for eight-inch disks shown in figure 6.*

The Disk Drive

The disk drive has four main functions:

1. To spin the disk at the correct rate;
2. To tell the controller when the index or sector marks have come around to the home position;
3. To place a read/write and erase head at the correct distance in from the disk edge; and
4. To translate the digital signals from the controller into the magnetic pulses on the disk (figure 8 shows a block diagram of these activities). Refer also to figure 9 as a guide to the parts of the drive discussed below.

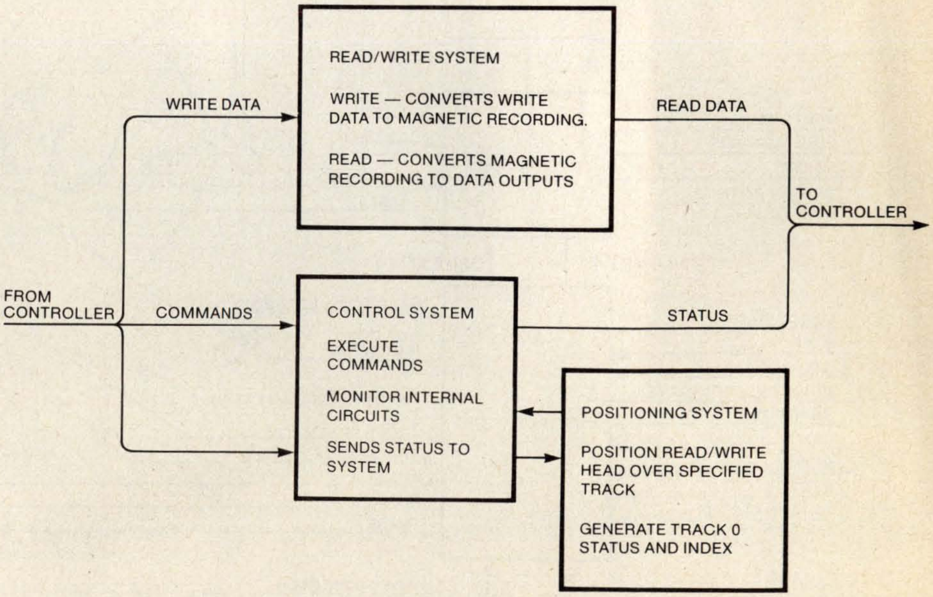

READ/WRITE SYSTEM
WRITE — CONVERTS WRITE DATA TO MAGNETIC RECORDING.
READ — CONVERTS MAGNETIC RECORDING TO DATA OUTPUTS
WRITE DATA
READ DATA
TO CONTROLLER
FROM CONTROLLER
COMMANDS
CONTROL SYSTEM
EXECUTE COMMANDS
MONITOR INTERNAL CIRCUITS
SENDS STATUS TO SYSTEM
STATUS
POSITIONING SYSTEM
POSITION READ/WRITE HEAD OVER SPECIFIED TRACK
GENERATE TRACK 0 STATUS AND INDEX

Figure 8. *The drive's task is to transfer the information from the controller to the disk surface. In addition to the read/write electronics, there must be a system for positioning the head to the right track and sensing the index hole, as well as various control systems to insure that all parts of the disk work with the right timing.*

Spinning the disk around is the easiest part. On five-inch drives, this is done by a DC motor, which runs off the disk power supply. Speed regulation can be electromechanical but is generally electronic, since the speed must be accurate to within 1%.

Eight-inch drives may use DC motors, but most use AC motors driven from the regular power line (this means that the motors must be different for 115 V 60 Hz used in North America and 230 V 50 Hz used in most of the rest of the world).

Detecting when the disk is at the home position is done with a photocell, which is placed to look for a hole in the disk which comes around once each rotation. This is the index hole, and on all soft-sectored disks this is the only small hole in the disk. However, the corresponding hole in the jacket, where the photocell looks through to find the index hole in the disk material is located at a slightly different position on one-sided and two-sided disks. This allows drives to automatically sense when a two-sided disk is being used by checking to see whether a photocell at the two-sided position is ever activated.

Some disks and controllers get confused if both holes (for one-sided and two-sided disks) are punched out on a disk. Since a disk must be formatted as one or the other at a particular time, the solution is to simply cover up the unnecessary hole with the opaque tape used to cover the write-protect notch. It is important to use a tape or sticker that doesn't exude extra adhesive onto the disk.

The read/write head is carried in and out on a holder called the *carriage*, which in turn is driven by one of several different methods. Most drives use a lead screw (a rod with very coarse threads, the turning of which drives the carriage back and forth like a nut on a turning bolt). Some use a band positioner, which employs a metal band running over a pulley. In both these cases, a stepper motor (one that moves in discrete jumps) "steps" the head in and out.

Another method of moving the head in and out is to use a voice-coil motor, which produces linear motion. This is the method employed on drives made by Persci, Inc. Although more costly, it does allow the head to move between tracks more quickly.

Writing data on the disk and reading it is done by a read/write head, which is a tiny electromagnet. The head turns pulses of current into magnetic changes on the disk in write mode, and magnetic changes into pulses of current when reading. A good part of the electronic parts in the drive are concerned with shaping the digital data from the computer into pulses of current, and changing the current pulses from the head into properly shaped digital outputs.

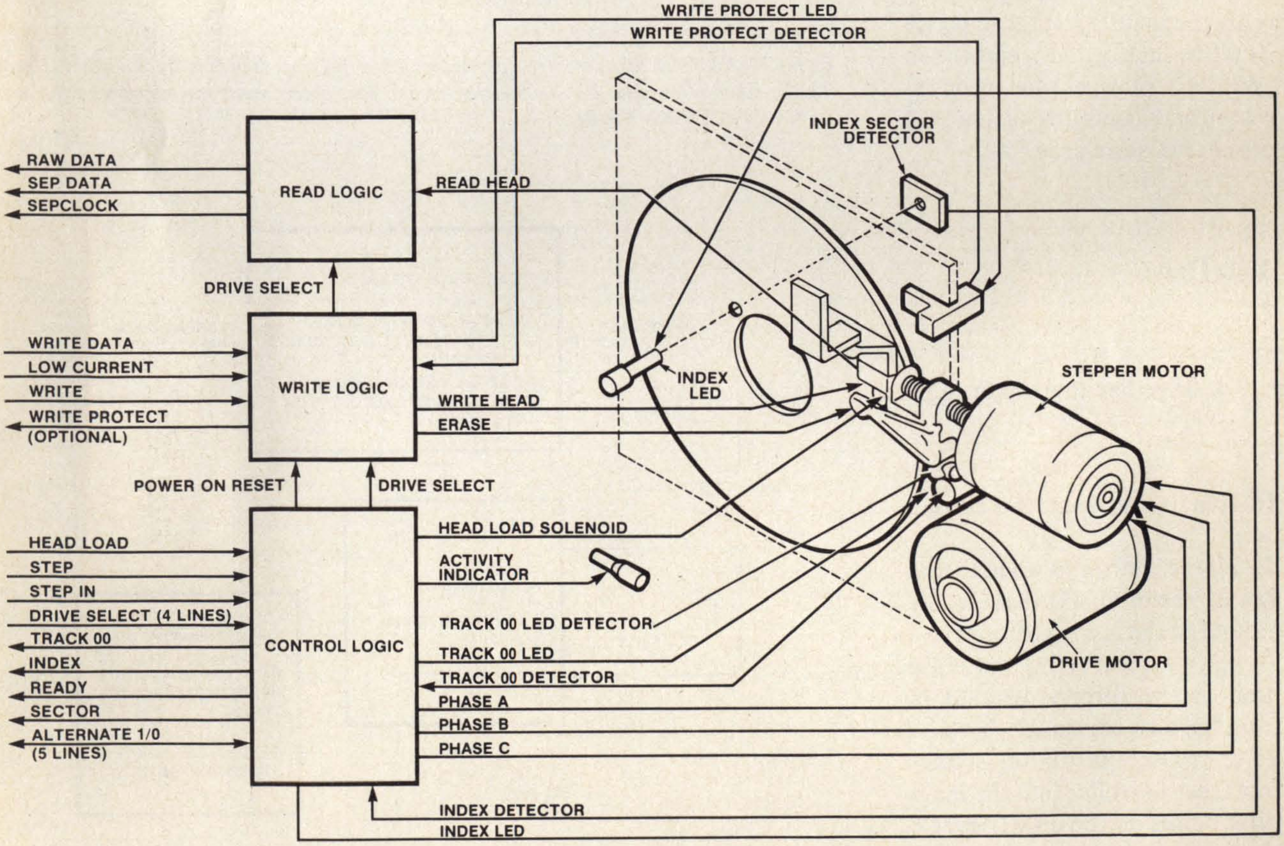

WRITE PROTECT LED
WRITE PROTECT DETECTOR
RAW DATA
SEP DATA
SEPCLOCK
READ LOGIC
READ HEAD
INDEX SECTOR DETECTOR
DRIVE SELECT
WRITE DATA
LOW CURRENT
WRITE
WRITE PROTECT (OPTIONAL)
WRITE LOGIC
WRITE HEAD
ERASE
INDEX LED
STEPPER MOTOR
POWER ON RESET
DRIVE SELECT
HEAD LOAD
STEP
STEP IN
DRIVE SELECT (4 LINES)
TRACK 00
INDEX
READY
SECTOR
ALTERNATE 1/0 (5 LINES)
CONTROL LOGIC
HEAD LOAD SOLENOID
ACTIVITY INDICATOR
TRACK 00 LED DETECTOR
TRACK 00 LED
TRACK 00 DETECTOR
PHASE A
PHASE B
PHASE C
DRIVE MOTOR
INDEX DETECTOR
INDEX LED

Figure 9. *Although the drives made by different manufacturers use different internal designs, the principles remain the same; this schematic diagram of the mechanism for a floppy disk drive is fairly representative.*

The Computer, the Drive, and The Controller

To use the floppy disk, we want the computer to act as our agent and put data onto the floppy (write data out to disk), or retrieve data from the floppy (read data in from disk). Traditionally, the in and out direction refers to the direction of data flow seen from the computer's point of view.

However, the computer itself was not designed to know all about different disk drives, and therefore cannot generate the required instructions in the form needed by the drive. The computer needs an added device, called a *controller*, which is usually a circuit card which sits inside the computer box and translates back and forth from the computer to the disk drive.

Controllers are fairly standard devices in their design. Most controllers use an integrated circuit to do the main tasks.

From our point of view, this means that we access information on the disk in a stylized fashion. If we have a multi-format controller, our software first sets the controller to the chosen format. Then, it tells the controller what track to look for, and which sector on that track. Then, it either presents data to the controller for writing on the chip or receives data which has come back from the disk.

This scheme of doing things is responsible for much of the architecture of CP/M. The complicated nature of it means that if we want to write our own programs which use disk storage, we can perform disk access in certain ways which are more practical than others, and be careful not to try operations that are impossible. In using programs which have been correctly written for CP/M, we can ignore most of the details, since they are taken care of by the CP/M operating system.

One controller can run several disk drives. Most controllers have circuitry for driving four or more, although most systems can do very nicely with two. A few controllers will support both five-inch and eight-inch drives at the same time, and a few others can be strapped to talk to one or the other but not both at once.

The Bootstrap

CP/M is designed to run while residing in the read-write area of memory (known as RAM, or random-access memory), and CP/M disappears when the power is turned off. Since it is CP/M itself which normally tells the computer how and where to read the disk and how to get other programs started, some means must be provided to tell the computer how to read CP/M from the disk first.

The process of entering in a small program which in turn reads in a bigger operating system program is called "bootstrapping" (from the old phrase of pulling oneself up by one's bootstraps), and the small program is called the "boot."

On older computers, the boot program was entered into the computer by changing the front-panel switches. On the latest computers, the boot program is usually provided in read-only memory (which doesn't change when the power goes off), either on the disk controller card or as a "monitor" program in the computer. The computer is set to execute this program at power-on or after the "reset" button is pressed. Without a bootstrap, the computer cannot run CP/M.

The Sounds of Disks at Work

Most of the time, the disk is just resting there, or going round and round inside its jacket, waiting for the drive to read or write something. Five-inch disks, which generally use DC motors run from the disk power supply, are often set to stop their motors when no data has been transferred for a while. The motors are started up again when a data transfer is requested (the startup usually takes one second). This makes a characteristic whirring sound, which to most people becomes quite reassuring since it seemingly indicates that the computer is doing something.

Most eight-inch drives run all the time the computer is on, creating a significant sound level (about equal to a desktop fan). Since eight-inch drives generally use AC motors driven from the power line, they are harder to start and stop (they also take about two seconds to get up to speed). Few drives of this size are made to turn off their motors when not in use.

On most models of both size drives, the recording head is kept away from the surface of the disk even when the disk is rotating, unless a transfer is underway or location information on the disk is being checked. However, some five-inch drives and a few thin eight-inch drives omit the head-load mechanism, and rely on turning off the motor between transfers to limit wear.

On models with head-loading mechanisms, an electromagnetic brings the head into contact with the disk surface upon request, making a click (five-inch) or thunk (eight-inch). On most drives, a small red light on the front of the drive also goes on, indicating the head is loaded. Some models have door locks which prevent the door from being opened while the head is down.

Lastly, the heads are moved in and out from the back of the drive towards the center of the disk, so that one set of heads can access all the tracks. This too has its characteristic hum, and eventually you will find you will often be able to tell when things have gone wrong from the change in sounds as the disk does something unexpected.

Disk Errors

Although floppy-disk and drive technology is now fairly debugged (at least for one-sided single-density drives and disks), a certain number of errors can still be expected to occur in the writing and reading processes. A small portion of data bits written as 1's will read back incorrectly as 0's, and vice-versa.

Most of these errors are detected immediately when the data is read because the disk controller adds a *checksum*, a number derived from a formula known as the cyclic redundancy check (CRC), to each block of data as it is written out. This checksum is a number calculated by running all the other bits in the block through a special mathematical function. Even a slightly different data pattern results in a different checksum.

When the controller reads a block of data into the computer, it calculates what the checksum of that data should be and compares it to the checksum on the disk. If the computed checksum and the checksum on disk do not agree, the controller signals the computer that the block just read was not correct.

Although every possible error will not be detected in this fashion, almost all common ones will, including virtually all of those caused by specks of dust or small defects in the disk's magnetic coating. In practice, disk data put through valid cyclic redundancy checks (CRCs) is likely to be more reliable than data in memory.

The CRC is designed to guard against two kinds of disk errors: soft errors, where the correct information is obtained if the disk is re-read, and hard errors, where a reasonable number of re-tries still fails to produce the correct information.

Most operating systems, including CP/M, will automatically retry any read or write operation which causes the controller to report an error. These systems notifying us with a system error message only after a given number of attempts have failed to produce the right result.

What Else Can Go Wrong With a Disk?

One major cause of problems with disk and drives is "crunching" the disk: failing to get the disk set properly in the drive before closing the door. Even slight misplacement can cause the edge of the center hole to be bent or torn, which may cause the disk to be off center. This in turn may cause data loss or marginal operation.

While most drives now have "anti-crunch" mechanisms, which keep the door from closing unless the disk is fully inserted, these are not foolproof. When edges are bent or torn, you can use reinforcing rings made of plastic, which are available from most computer supply houses to repair the damage. Some people put them on as preventive measures, but they may not be effective against "crunching."

The surface of a disk is rarely worn out by a properly-functioning drive. Most disks are rated for tens to hundreds of hours of head contact, which typically means many thousands of hours of use (since the head is in contact with the disk only during reads and writes).

However, a drive which fails to unload the head when not reading or writing will wear both head and disk out in a matter of perhaps hours or days. Disks should not be left running unattended for long periods of time if this seems to be a significant possibility.

If you use a single-drive system, you may cause more wear and tear on your disks as you do the "floppy shuffle" in disk-to-disk copying. In every case, handle disks with the care you would have if you were holding a 250,000-character draft of a prize-winning novel. Make copies of all disks, and may you never lose a single character.

MAXELL FLOPPY DISK CROSS REFERENCE

TYPE	PRODUCT DESCRIPTION	MAXELL	SHUGART	IBM	DYSAN	3M	VERBATIM	MEMOREX	WABASH
Eight Inch Single Sided Soft Sectored	Single Sided Single Density 128 Bytes	FD1-128-M-1100 FD1-128-M-1200*	SA100	2305830	3740	740-0	FD34-1000, 3718 (FD34-9000, 3714 Critical Certification)	3060	F111111
	Single Sided Double Density 256 Bytes	FD1-128-M-1100 FD1-128-M-1200*	SA102	—	3740D	741-0	FD34-8000 FD34 -4495	—	—
Eight Inch Single Sided Hard Sectored	Single Sided Single Density 32 Sectors	FH1-32-M-1200*	SA101	—	101S	740-32	FD32-1000, 3716 (FD32-9000, 3713 Critical Certification)	3015	F31A211
	Single Sided Double Density 32 Sectors	FH1-32-M-1200*	SA103	—	101D	741-32	FD32-8000 FD32-4494	3091	F33A211
Eight Inch Double Sided Double Density Soft Sectored	26 Sectors 256 Bytes	FD2-X-DM-1100† FD2-X-DM-1200*†	SA150	1766872	800/850	743-0 (256)	DD34-7219	3103	F142111
	15 Sectors 256 Bytes	FD2-X-DM-1100† FD2-X-DM-1200*†	—	2736700	—	—	FD10-4015	3106	F122111
	15 Sectors 512 Bytes	FD2-X-DM-1100† FD2-X-DM-1200*†	—	1669044	800/850	743-0 (512)	DD34-8247	3114	F145111
	8 Sectors 1024 Bytes	FD2-X-DM-1100† FD2-X-DM-1200*†	—	1669045	800/850	743-0 (1024)	DD34-7218	3104	F147111
Eight Inch Double Sided Double Density Hard Sectored	32 Sectors 256 Bytes	FH2-32DM-1100 FH2-32DM-1200*	SA151	—	800/851	743-32	DD32-7011	3105	F34A411
5¼" Mini Single Sided Single/Double Density	Soft Sectors	MD1-M	SA104	—	104	744-0	MD525-01 (4443)	3401	M11A111
	10 Hard Sectors	MH1-10M	SA107	—	107	744-10	MD525-10 (4445)	3403	M41A111
	16 Hard Sectors	MH1-16M	SA105	—	105	744-16	MD525-16 (4444)	3405	M51A111
5¼" Mini Double Sided Single/Double Density	Soft Sectors	MD2-DM	SA154	—	104/2D	745-0	MD550-01 (6738)	3421	M14A-211X
	10 Hard Sectors	MH2-10DM	SA157	—	107/2D	745-10	MD550-10 (6740)	3423	M44A-211X
	16 Hard Sectors	MH2-16DM	SA155	—	105/2D	745-16	MD550-16 (6739)	3425	M54A-211X
5¼" Mini 96 TPI Soft Sector	Single-sided	MD1-DDM (96 TPI)	SA410	—	204/1D	—	577-01	—	—
	Double-sided	MD2-DDM (96 TPI)	SA460	—	204/2D	—	557-01	—	—

†FD2-X-DM-1100 and FD2-X-DM-1200 are uninitialized and can be formatted to any byte configuration (256, 512 or 1024).
*Denotes Write Protect Feature

TYPES OF MAXELL FLOPPY DISK COMPATIBILITY AND CROSS REFERENCE INFORMATION

		Single/ Double-sided	Recording Density	Bytes Sector	Sectors/Track [S/T] Sector Holes [SH]	Capacity [K bytes]	Maxell	IBM	Shugart
8-inch Floppy Disks	Soft-Sectored	Single	Single	128	26 S/T	243	FD1-128*	2305830	SA100
				256	15 S/T	280		2305845	
				512	8 S/T	299		1669954	
			Double	256	26 S/T	486			SA102
				512	15 S/T	561			
				1,024	8 S/T	598			
		Double	Single	128	26 S/T	493		1766870	SA150
				256	15 S/T	568		2736700	
				512	8 S/T	606			
			Double	256	26 S/T	985	FD2-XD	1766872	
				512	15 S/T	1,137		1669044	SA150
				1,024	8 S/T	1,212		1669045	
	Hard-Sectored	Single	Single		32 SH	315	FH1-32*		SA101
			Double		32 SH	631			SA103
		Double	Single		32 SH	631			SA151
			Double		32 SH	1,262	FH2-32D		
5¼″ Mini Floppy Disks	Soft-Sectored	Single	Single	128	16 S/T	82	MD1		SA104
			Double	256	16 S/T	164			
		Double	Single	128	16 S/T	164			
			Double	256	16 S/T	328	MD2-D MD1-DDM (96TPI)		SA154
	Soft-Sectored	Single	Double	256	16 S/T	164			
		Double	Double	256	16 S/T	328	MD2-DDM (96TPI)		
	Hard-Sectored	Single	Single	128	10 SH	82	MH1-10		SA107
			Double	256	10 SH	164			
		Single	Single	128	16 SH	82	MH1-16		SA105
			Double	256	16 SH	164			
		Double	Single	128	10 SH	164			
			Double	256	10 SH	328	MH2-D-10		SA157
		Double	Single	128	16 SH	164			
			Double	256	16 SH	328	MH2-D-16		SA155

*Meet ANSI single and double density specifications.

The CompuPro S-100 Computer

Near-Excellence in Hardware From Bill Godbout's Oakland Airport Hangar

by Allan Lundell

CompuPro computers are a unique breed of machines, created by a company that has an outstanding reputation for quality, reliability and technical innovation.

Started by Bill Godbout in 1974, CompuPro has maintained its small but unique facility near the Oakland Airport in the San Francisco Bay area. Bill is one of the original entrepreneurs and engineering geniuses of the microcomputer revolution and has a dynamic personality both feared and loved throughout the industry. His computers are but an extension of his personality, and are considered the best in their categories.

Hardware

CompuPro machines are not considered personal computers. They are more expensive and more sophisticated than the majority of small computers on the market today — expect to pay at least $10,000 for a CompuPro system. To Bill, the personal computers on the market are almost a joke, hardly suitable for real business and professional applications.

It is a bit ironic that his machines use the "S-100 Bus," a famous personal computer standard from the 70's that has been refined by Bill and his team from its original status as a hobbyist standard to a polished level of sophistication and performance.

"S-100 Bus" is the name given to the hardware connection, called a *bus* or *buss*, that connects various circuit boards in your computer to form a computer system. These circuit boards may contain central processing units (CPUs), memory (RAM or ROM), device drivers (for added disk drives, hard disks and other such devices) and modem boards (for communication over phone lines).

The S-100 Bus enables many products not made specifically for CompuPro to be used in the computer. Hundreds of computer companies have created products for S-100 machines, giving considerable latitude to the potential applications of an S-100 machine.

The CompuPro computer enclosure is very solid, very metallic, well-ventilated, extremely rugged, almost militaristic — and it is tough, capable of existing almost anywhere.

CompuPro did not stop with a good bus in a box. CompuPro has not depended on other manufacturers to create S-100 hardware cards for their machine. Rather, they have created dozens of plug-in S-100 boards themselves, some with very unique state-of-the-art capabilities.

Although you can walk into certain stores (such as G&G Engineering outlets) and buy a CompuPro ready-made to work in your application, you can also have one configured with the capabilities you want. With plug-in S-100 cards, this is a fairly straight-forward process, though it does require someone who knows how to configure the system. Similar to equiping an automobile with accessories, this personal configuration process helps give your computer a personality based on what you want.

The system we have for review was configured by Bill himself. It has the capabilities required for any small business — plus a few added goodies. We have nicknamed our CompuPro system "Albert" in honor of our well-known relativistic friend. (*Ed. note: not to be confused with the new Albert computer, an Apple-II lookalike introduced at the Atlanta COMDEX show in late April*).

Our system configuration includes the following CompuPro components:

The Enclosure 2, desk version — basically, the box and power supply. It has a 20 slot S-100 motherboard, constant voltage power supply, line filter, positive pressurized filtered ventilation system, and "rugged all-metal construction."

The CPU 8085/88 — A dual processor (two CPU) board which runs both standard eight-bit (8085) and sixteen-bit (8088) software. This board lets us run both existing and future software packages that may be developed for the 8088 processor (the same used in the IBM Personal Computer and PC lookalikes). This board is one of CompuPro's most successful boards, although Bill had not anticipated its success. He happened to be in the right place at the right time with the right board for our slow transition from eight-bit to sixteen-bit computer systems.

The RAM 16 — Two RAM 16 low power, high speed, high density (64K byte) memory boards.

The Interfacer 3-5 — an eight-channel RS-232 serial input/output board that is ideal for multi-user environments. Four of these boards would provide multi-terminal use for up to 32 users while occupying only eight physical port locations. The board allows fast data transfer among a wide variety of different computers and computer peripherals, such as CRTs, modems, and printers.

System Support 1 — A single board using LSI technology that has the most popular system support functions: dual interrupt controllers, three 16-bit interval timers, EPROM sockets, real time/clock/calendar, full RS-232 serial channel, a power failure warning signal, and an unusual memory enable-/disable function called PHANTOM control.

Disk 1 — an LSI-based floppy disk controller based on the third generation INTEL 8272/NEC 765A floppy-disk controller IC. It can handle up to four eight-inch or 5.25-inch drives, single or double density, single or double sided.

256K M-Drive — This is a "solid-state disk drive" or RAM drive, drastically increasing system speed and performance. More later on this handy feature.

Two Qume double-sided, double-density Trak 842 eight-inch floppy drives, a Televideo 950 terminal and Diablo 630 letter quality printer complete the system.

Software

CompuPro systems are capable of running the majority of eight or 16 bit operating systems. The most popular are CP/M-80 (eight-bit), MP/M (eight and 16-bit versions), CP/M-86, Oasis, and Oasis 16, in that order.

CP/M (and MP/M) offer the most comprehensive business-oriented software available to date, including such programs as WordStar and SuperWriter for word processing, SuperCalc-86 for business and financial planning, and dBase II for data base management. CompuPro's floppy drives can read and write single-density CP/M standard disks (the IBM 3740 standard), allowing data on a CompuPro machine to be easily transferred to any other machine that reads CP/M single-density disks.

Reliability

Amazing. Over the last year, Albert has been on the road without a single failure. He has even served as the "unofficial" (and only) word processor-/computer at the site of Steve Wozniak's US festival (known by those who lived in the on-site trailers in Devore, California as the Dust Festival).

A few days ago, I attempted to demonstrate my PMMI (Potomac Micro-Magic, Inc.) "intelligent" modem to a visiting friend. Albert booted up my Modem7 program without a hitch, but the PMMI modem did not detect a dial tone. Puzzled, I took the lid off Albert and discovered that the ventilation fan was jammed by its own air filter!! Who knows how long Albert had survived without air...

The CompuPro S-100 boards were very hot to the touch, yet all were working perfectly — except for the PMMI modem board. Closer examination of the boards shows the superior craftsmanship of the CompuPro boards. For example, CompuPro uses IC (integrated circuit) sockets for all integrated circuits, rather than soldering the IC's directly onto the circuit board. This simplifies maintenance and allows more dissipation of heat. To be fair, however, the PMMI S-100 board worked fine after cooling down for a few minutes.

Aesthetically however, Albert is still a box. The newer enclosures have a horizontal "racing" stripe on the front panel, but they are still candidates for the military. The design is dull, yet simple, effective, and straightforward.

CompuPro has not left style out of the picture in terminal selection, however. CompuPro is recommending the new QUME terminal, a spacy design right out of the future (next year). Very stylish, the terminal incorporates many user-oriented features, including detached keyboard, tilted green screen, keypad, and software-defined special function keys. A nice choice. Albert, however, uses the Televideo 950, also a nice terminal, but bulkier and last year's model.

The dual processor (The CompuPro 8085/8088 board) is perhaps the best of its kind. It gives you the capability to run both eight-bit and 16-bit software. The eight-bit 8085, however, does not run some software designed for the Z-80 chip, making it important to check your software for chip compatibility before buying. Nearly every CP/M program will run.

Though we have primarily used CP/M-80 as our operating system, CP/M-86, MP/M, MS-DOS, Oasis and Oasis-16 are also available. This gives users an incredibly large selection of software and special capabilities. We are planning a review of the CompuPro handling multi-terminal operating systems and software in a future issue of *User's Guide*.

Memory Disk Drive

One of Albert's most valued features is the M-Drive. Known to some as the Warp Drive, the M-Drive looks just like a disk drive, but is actually a portion of RAM (random access memory). Data access occurs at the speed of internal memory, without the use of

The CompuPro® 816/C® multi-user system with futuristic Qume terminals.

disk drive mechanisms, increasing system performance by as much as 3500%! Waiting time between steps of a program is virtually eliminated if the program is loaded into the M-Drive.

You notice the difference immediately if you use a program like WordStar. Saving a long file can be particularly annoying using floppy disks, especially when you are waiting for the drives to finish before you can write your next inspired thought. The M-Drive can hold your file temporarily so that file saves can occur nearly instantly. Of course, you must eventually save the file on disk, since the M-Drive memory is erased when the power goes off.

Some users take advantage of the reliability of internal memory and use a battery backup to keep the RAM alive. If a power failure occurs during editing, and your system has a battery backup power source, the programs and files in RAM are still alive and in working order. Since power spikes (which can occur anytime) can cause "glitches" on the disk, the file stored in battery-powered RAM is actually safer!

Noise

Albert's constant hum is reasonably quiet. The sound is generated by the fans in the disk and motherboard enclosure. I sometimes catch myself humming along with them, creating a rather pleasant harmonic. The hum occasionally changes pitch for some reason, scaring me into saving my file immediately.

I live in the Santa Cruz mountains, and power failures are a way of life here. All too often I have had no warning of an impending power outage, reducing my latest article from a potential cover story for *Newsweek* to a dull phosphorescent after-image on a powerless CRT. Such is life in the mountains...

I had one other noise problem of a different sort. During Albert's stay in the town of Bonny Doon, he became rather infamous for his ability to destroy television reception — particularly with neighbors who were connoisseurs of quality TV.

I tried every imaginable shielding technique for radio frequency, from RF chokes to grounding Albert to a sunken copper pipe, and only succeeded in reducing the overall noise. To be fair, however, Albert only succeeded in destroying reception on a few stations, notably 7 and 10. RF interference is usually not a problem for business users. In fact, I have seen TV sets operating at CompuPro with little or no RF interference (perhaps the electrical wiring in their building is better than Albert was used to in Bonny Doon).

Simple or Complex?

Albert is extremely simple to turn

on. A large red rocker power switch on the front panel turns the entire system, including the printer, on or off. Three power switch-controlled electrical sockets are on Albert's back panel, similar to a good stereo system. These sockets are very handy for plugging in the terminal, printer, disk drives, or other computer peripherals that only need power when the main system is on.

A red-lit Reset button, located above the main power switch on Albert's front panel, is very handy for restarting the system. If, for some reason (perhaps static electrical discharge), the keyboard "locks up" or "freezes," pushing the reset button will get you out of it. You may lose any data entered before the last "save," but at least the computer is usable again.

What is truly unusual are the times Albert freezes. This doesn't happen as much up here in the mountains as when Albert and I lived in the town of Santa Cruz. Every so often, without warning, the keyboard would "die" (stop working). I suspect that the Televideo terminal, not the CompuPro, is the cause of this mystery. Often the system works after turning the terminal itself off, and then on.

From my experience with other computers as well as Albert, behavior may change in different locations as a result of such things as power surges, brownouts, software glitches, cosmic rays, weather, synchronistic thoughtforms, a combination of events, etc. Albert performed operations problem-free 99 times out of 100. Can you say the same about your car?

Documentation

Bill Godbout does his own manuals. He's got his own everything at his hanger facility at the Oakland Airport, including printing and photographic reproduction capabilities.

But this does not mean that the manuals are user-simple. They are detailed, contain a ton of information on every technical point worth mentioning, and are extremely useful — to other engineers.

CompuPro equipment is not really competing in the personal computer market. CompuPro computers are oriented to the business and professional marketplace, carving out a niche normally filled by the larger and more expensive "minicomputers." As a result of this approach, CompuPro computers are usually installed by trained dealers who understand the computer without the need of a manual oriented to beginners.

That means beginners must look elsewhere for tutorial information. CompuPro systems run CP/M and standard programs that are described in other tutorials (such as *User's Guide*).

Serviceability and Setup

CompuPro has over 2 dozen full service system centers in North America, with headquarters at the Oakland Airport, California. They offer full hardware and software support, system installation, and training.

At no extra cost, CompuPro offers a one year warranty (from date of purchase of end user) on the boards, mainframe enclosures, and floppy disk drives. The people behind CompuPro equipment are proud of their products and it shows in their marketing and servicing.

Bill and his engineering team put together Albert for the first time, checked him out, labeled the cables, and ran numerous hardware and software tests. However, we've taken him apart many times during his numerous moves, and have never had any trouble putting him back together. Assembling a CompuPro system, once it has been configured, is a piece of cake. It helps to have the cables labeled properly.

Summary

CompuPro makes good hardware. Few companies create products of comparable craftsmanship. In many ways, the CompuPro system is the Rolls Royce of the microcomputer industry. CompuPro machines cost a little more, but pay off in consistent, dependable service. Our hats are off to Bill and his team for establishing such a high standard in technological excellence.

Attention CB80 & CB86 users:

Here's a field editing and screen management utility that lets you create interactive software in a fraction of the time for a fraction of the cost!

- The Interactive Program Aid (IPA-1) for compiled CBASIC under either CP/M-80 or CP/M-86
- Source and detailed manual for modifying to your special needs
- Quickly and easily displays data fields and literal strings
- Edits for data type (alphanumeric, numeric and numeric strings) minimum, maximum lengths, dates and Y/N
- Allows editing while data is being entered
- Fields and literal strings displayed in normal or high-lighted intensity
- Cursor control features include left-right movement, delete, insert and skip
- Field termination by carriage return or programmer defined function keys
- Virtually terminal independent...works on almost any terminal with addressable cursor

$30⁰⁰ plus $5.00 mailing and handling

(IPA is furnished on 8" SSSD, inquire about availability for other media)

Compare to programs costing ten times more!
Put this creative computing innovation at your fingertips...fill out and mail this coupon with remittance today.

COMPUTER THAUMATURGY

Problem Solvers and Consultants to Computer Users

Suite 700 Miami Valley Tower
40 W. 4th Street • Dayton, Ohio 45402

☐ Please send the IPA-1 user's guide (enclose $5.00 to cover costs and handling and mailing)
☐ Please send the IPA-1 Programming Utility (Postage and Handling included)
Enclosed is my check for $35.00
Ohio residents add 6% sales tax

name ____________________

company ____________________

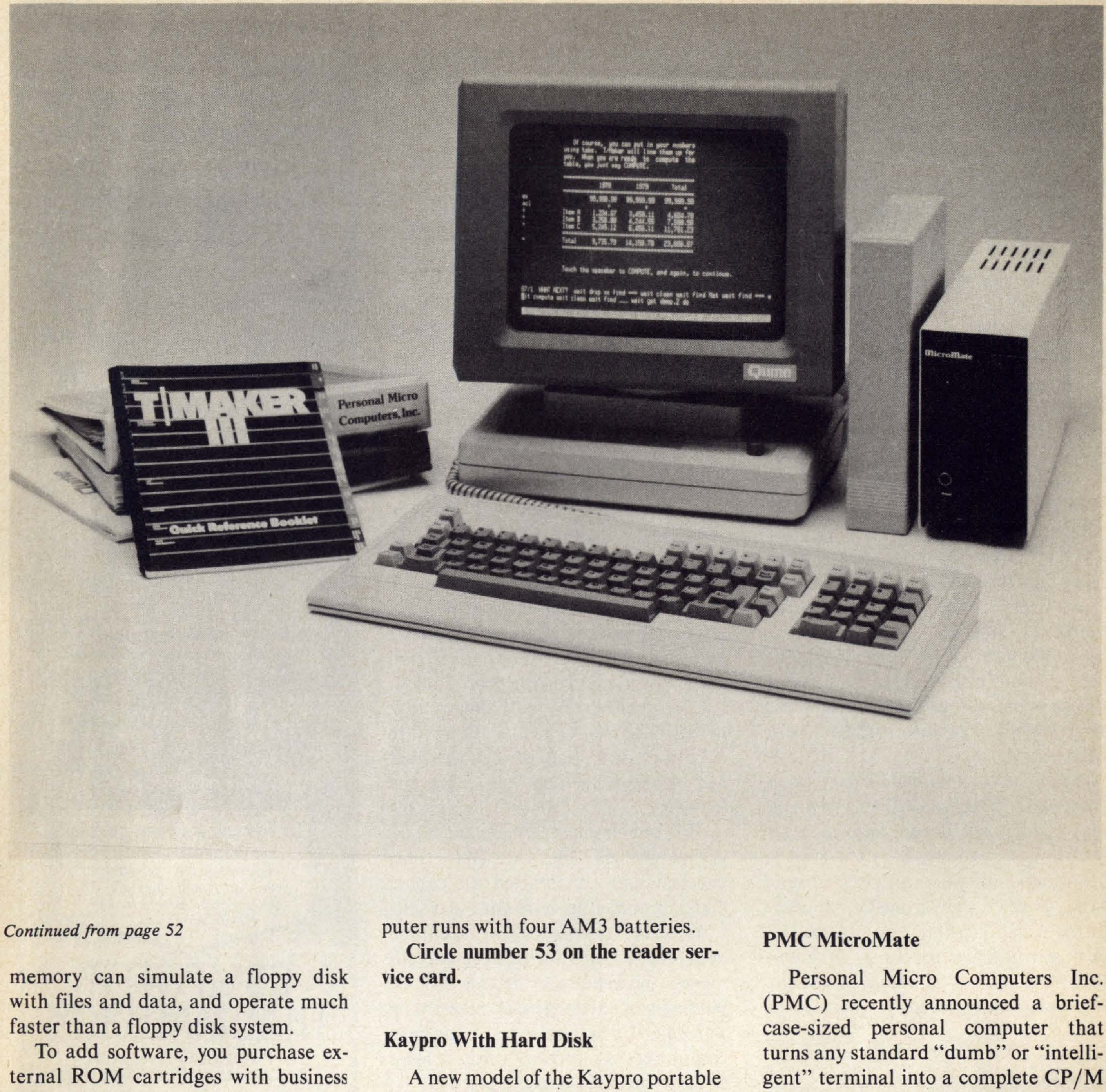
address ____________________

city ____________________

state ____________ zip ____________

CP/M, CP/M-86, CBASIC, CB80 and CB86 are registered trademarks of Digital Research.

Circle reader service number 4



Continued from page 52

memory can simulate a floppy disk with files and data, and operate much faster than a floppy disk system.

To add software, you purchase external ROM cartridges with business software, utilities, games, etc. To store data, you use external RAM cartridges, add external floppy disk drives, or telecommunicate the data to other computers. The unit does not run CP/M, but the ASCII text and data files can be transferred to nearly every computer system.

The unit has a standard 67-key keyboard, an eight-line by forty column display (LCD) with full alphanumeric and graphic characters, a printer port, RS232 port, and cassette port. AC power adapter is an option, along with a rechargeable battery pack; the computer runs with four AM3 batteries.

Circle number 53 on the reader service card.

Kaypro With Hard Disk

A new model of the Kaypro portable computer includes a 10 megabyte hard disk with one 5.25-inch floppy disk (400K). The unit weighs 31 pounds and includes a standard typewriter-style keyboard with 72 keys including 20 programmable keys, 14-key numeric keypad, and four cursor keys. The high resolution nine-inch green phosphor display screen can display both text and graphics. The Kaypro also has one parallel Centronics-type printer connector and two RS232C serial connectors. Price: $2800.

Circle number 54 on the reader service card.

PMC MicroMate

Personal Micro Computers Inc. (PMC) recently announced a brief-case-sized personal computer that turns any standard "dumb" or "intelligent" terminal into a complete CP/M computer system. The MicroMate price/performance is excellent: $1095 for the computer without a terminal (you provide your own), or $1790 for an excellent package including the new Qume QVT-102 terminal, CP/M Plus and T/Maker III (the integrated word processing/spreadsheet package that provides list processing, file management, communications, bar graphs, etc.). The unit comes with 128K of internal memory, two RS232 serial connectors (terminal, modem, etc.), an eight-bit parallel connector for system expansion, a Centronics-type parallel

Advertise in issue 5
of *User's Guide.*

Reach 25,000 users of CP/M systems and software!
Issue will be at the IBM PC Faire.

For a media kit with current advertising rates, please contact: Judy Blake, **Hajar Associate** (415) 964-0706, or **User's Guide**, P.O. Box 3050, Stanford, CA 94305.

CUTE©
by
User Friendly Software®

Client writeUp sysTEm
for the accounting professional

CUTE© — easily installed, easy to use — brings your practice up to a high level of efficiency. Self-verifying data entry makes trial balance unnecessary, and you can trust non-professional assistants to process clients with this User-Friendly Software familiar posting and entry techniques.

Multi-user system includes General Ledger, verified bank reconciliation, comparatives, budget and post-facto payroll reporting — in standard charts and reports which are easily customized to your own formats.

- Self-prompting — menu-driven
- Standard operating system: (C/PM, M/PM) runs on most micros
- Written in PL/1

To find out how easily and inexpensively you can streamline your practice with CUTE, call or write:

User Friendly Software®
a Division of
ALPHA MERICS, INC.
1625 South 21st Avenue • Hollywood, Florida 33020
Dade (305) 949-8319 • Broward (305) 925-8194

Circle reader service number 1

connector (for a printer), and a connector for up to three additional 5.25-inch disk drives.

Circle number 55 on the reader service card.

Local Networks:

EXO Avalanche System

EXO Corp. introduced its Avalanche Networking Computer Systems with EXO's proprietary EXO/NET® local area network for office automation. Avalanche computers can function as stand-alone systems and can also be linked into a local EXO/NET network for sharing hard disks and printers, sharing data files, and performing electronic mail. All Avalanche systems run CP/M. The Model 200, at $1295, includes a single disk drive with half a megabyte; the model 300, at $1795, has two disk drives. The model 100, at $795, can upgrade any terminal into a CP/M-based workstation on the network.

Avalanche model 500 ($5995) includes a 20 megabyte Winchester hard disk, and the model 700 ($7995) includes a 46 megabyte hard disk. Both can function as EXO/NET file and printer servers.

EXO/NET, which uses baseband bus topology with CSMA/CD for collision detection, can connect one network server to up to 31 additional workstations or servers. Sixteen users can operate on the network at any one time. All workstations and file servers operate independently and are completely software and hardware compatible, so that a workstation can be turned into a file server, avoiding any delays with a server failure.

Circle number 56 on the reader service card.

Mass Storage:

Corvus Removable Tape "Bank"

Corvus Systems announced an inexpensive mass data storage device that can hold 200 megabytes on a removable cartridge. This is a major breakthrough in archival media. Called "The Bank," the device is approximately the same size and price as a 5.25-inch Winchester hard disk: $2200. The media is a continuous loop of 100-track magnetic tape encased in a 5.5-inch square by 1.5-inch thick cartridge.

The Bank's 200 megabyte cartridge takes an average of 10 seconds to find data. Once found, data is transferred at speeds up to 60K bytes per second. The storage device is ideal for local networks such as the Corvus Omninet.

Circle number 57 on the reader service card.

Applications Software:

Accounting:
Star Software Systems

Star Software offers accounting packages down-scaled from larger computers to work on personal computers. Comparable with programs priced above $3000, the Star Software package is an economical buy at $400 for the eight-bit CP/M version, and $500 for the sixteen-bit version (CP/M-86).

Circle number 58 on the reader service card.

Accounting: The Champion

The Champion, from Data Base Research (DBR) Corp, is a complete accounting system written in the dBASE II command language and bundled with a portion of dBASE II (the "runtime" software). You can purchase dBASE II separately and integrate The Champion with other dBASE II applications on CP/M systems.

The Champion includes general ledger, accounts receivable and payable, inventory and payroll, with a "help" module with answers for question marks you type while using the system.

Circle number 59 on the reader service card.

Data Base Mgmt.: DataFlex

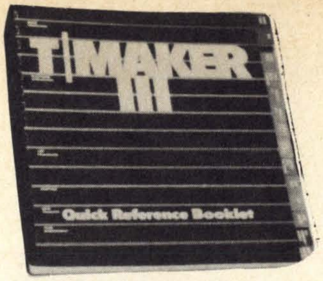
DataFlex, from Data Access Corp., is a relational data base management system with a programming language and application utilities. Features range from automatic file definitions to user-defined macros, with "help" messages on-screen. Applications already developed (from other sources) include Payroll, Pharmacy, Legal Billing, Architects Job Costing, Inventory, General Accounting and others.

Circle number 60 on the reader service card.

Data Base Mgmt.: DBPACK-II

DBPACK-II from Compu-Draw is a relational data base and information management system with extensive data entry, query, report generation, tabulation and formatting capabilities, with variable-length records and no limits on the number of files open at the same time, nor any limit on the number of keys for hierarchical sorting or selecting records. Password-protection is available, and data files are encrypted for greater security. On-line "help" messages are available at any time during operations. Price: $395.

Circle number 61 on the reader service card.



Integrated Packages: T/MAKER III

T/MAKER III is a word processing-based software package that performs a variety of things, including spreadsheet calculations, list processing, bar charting, data transfer and file management. All data (text and numbers) are "live" at any time to perform any of these functions. T/MAKER III is a unique yet easy-to-learn program whose predecessors (T/MAKER and T/MAKER II) received rave reviews. *User's Guide* plans to include an article on T/MAKER III in a future issue. T/MAKER III is available for CP/M, CP/M- 86, MSDOS and PCDOS.

Circle number 62 on the reader service card.

Decision-Making: Linear Programming

Digital Marketing Corp. announced LPMASTER, a program using linear programming methods to create a mathematical model of a project or an entire company. Linear programming models can help in determining the bottom line, in maximizing profits, in optimizing operations, exploring marginal alternatives, allocating scarce resources, presenting finished reports and minimizing capital expenditures. LPMASTER is $495 and runs on CP/M or MSDOS.

Circle number 63 on the reader service card.

Spreadsheets: Financial Templates

Simple Soft, Inc., introduced a line of templates, or spreadsheet models, called Quikcalc, available for VisiCalc® (VisiCorp) or SuperCalc® (Sorcim) spreadsheet programs on a variety of disk formats. Included are a loan analyzer, a depreciation analyzer, and a real estate investor.

Loan Analyzer provides detailed analysis of loan and mortgage instruments (suggested retail $100). Depreciation Analyzer handles any depreciation method including the new ACRS guidelines (suggested retail $100). Real Estate Investor presents data to evaluate both individual residence and income property sales and purchases (suggested retail $130). The Real Estate Investor has been endorsed by real-estate professionals.

Circle number 64 on the reader service card.

Statistical Package: SL-MICRO

The Questionnaire Service Company (QSC) announced SL-MICRO, a statistical language for micros that should already be familiar to many users of statistical languages running on larger mainframe computers. Included with SL-MICRO is a data conversion utility to make it compatible with data base management systems.

The statistical procedures include frequencies, crosstabs, condescriptive, Pearson correlation and multiple regression. SL-MICRO runs on any Z80-based computer running CP/M with a minimum of 48K internal memory. Price: $250. CP/M-86 version also available (for IBM PC or other disk formats) — 128K recommended, and CBASIC-86 required.

Circle number 65 on the reader service card.

System Utilities:

SmartKey From Heritage Software

SmartKey, from Heritage Software, lets you redefine any key on your keyboard, even while a program is executing. A key can represent another character, a word, a phrase, or a set of commands. SmartKey hides in a portion of memory below CP/M and does not affect program operation. It can be very useful in word processing — a single key can represent a name or an entire phrase to be repeated. Screenwriters can assign keys to store commands to change margins from screen directions to dialogue. SmartKey has been received for review and

Continued on page 82

A sampling of what you can expect . . .

Issue #1
The World of CP/M
Learning to Use CP/M
Impatient User's Guide to CP/M
CP/M Quick Reference and Summary
Using CompuServe
Preventing Computer Disasters
The Osborne Portable Computer
The IBM Personal Computer
The Alspa ACI-1

Issue #2
For Starters: Word Processing and Spreadsheet Calculations
A Guide to Using WordStar® (Pt. 1)
WordStar Impatient User's Guide
WordStar Installation Guide
Disk Doctor: The Rx For Bad Disks
CP/M Tutorial: Modifying Your BIOS, Files and the Directory
A Guide to Using SuperCalc® (Pt. 1)

Issue #3
For Starters: Mailing Lists
A Guide to Using MailMerge®
MailMerge Impatient User's Guide
Public Domain CP/M Software
Indexing Documents with DocuMate®
Footnoting Documents With FOOTNOTE®

User's Guide focuses on business applications for CP/M systems. The following articles are planned for future issues (this list is subject to change):

Using SuperCalc (Part two): Personal Income Taxes
Impatient User's Guide to SuperCalc
Using WordStar (Part two): Advanced Editing and Printing
Comparing the Portable Computers: Support and Performance
Public Domain Software: MODEM7, WASH
CP/M "Shell" Programs: CP+, Power, others
For Starters: Understanding Languages
Overview of CP/M Languages
Comparing Pascal Language Implementations
Introduction to PL/I and C
IBM Personal Computer with Concurrent CP/M-86
Overview of CP/M Hard Disk Subsystems
Public Domain Software: Bulletin Board Systems
CBASIC, CB-80, CB-86: Applications
Programmer's Helpers: Display and Access Managers
For Starters: Data Base Management Systems
Designing a Database With dBASE II
Comparisons: FMS-80, dBASE II, others
DIALOG: A Researcher's Database
Introduction to The Source
Accounting Plus: Business Management
Financial Planning: FPL, T/MAKER III, others
Introduction to Multi-User Systems and Local Networks
The Bottom Line Strategist

You will have to buy your own User's Guide.

If you are one of the 700,000+ users of CP/M systems and software, and you don't have a copy of *User's Guide* #1, order back issues today! Our first printing of 10,275 copies means that only one in 70 CP/M users will receive a copy of this collector's item. Use the subscription card in this issue to order at $4 each.

If you are interested in article reprints, write to: *User's Guide* Reprints, P. O. Box 3050, Stanford, CA 94305.

User's Guide
TO CP/M SYSTEMS & SOFTWARE

Continued from page 80

will be described in a future issue of *User's Guide*

Circle number 66 on the reader service card.

Menu/Power! and POWER! From Computing!

Computing! announces Menu/Power!, a multi-utility package designed for beginners that lets you manipulate files and perform disk "housekeeping," for $149. Also, Computing! announced an enhanced version of its POWER! utility that provides the same utilities in Menu/Power! plus more powerful utilities to read and write physical disk sectors, use passwords to protect access to files, and perform more advanced functions. The enhanced POWER! utility is $169 (an update from an earlier version of POWER! is only $35). POWER! has been received for review and will be described in a future issue of *User's Guide*

Circle number 67 on the reader service card.

Menu Master

Borland International introduced Menu Master, which is a menu-style file management utility for CP/M. Menu Master lets you control "job streams" — opening and closing files, using multiple commands and passing parameters to running programs — and thereby perform complex operations by pressing a single key.

Menu Master consists of a general-purpose set of menus for typical operating system functions. Also included are a file menu, word processing menu, and a development menu for creating customized menus. Menu Master provides a "programming language" for job control called MDL, which is syntactically similar to Pascal or BASIC. Price: $195.

Circle number 68 on the reader service card.

Disk Fix

The Software Store introduced Disk Fix, a disk repair utility for CP/M systems. Disk Fix can display, change and copy any sector of a CP/M 2.2 floppy disk or hard disk, and can recover erased files even after a directory malfunction. Disk Fix can also help reconstruct files with bad sectors.

Circle number 69 on the reader service card.

Filer

Filer, from Compu-Draw, is a program for compacting, archiving and cataloging disk files in CP/M systems. In applications with a lot of small files, Filer can offer disk space savings of several hundred percent. Filer also provides the ability to stamp files by time, date and descriptive text. Filer can organize up to 1000 files in one volume. Updates of files in the same volume are automatically assigned version numbers. Price: $49.

Circle number 70 on the reader service card.

Information & Videotex:

New Services On CompuServe

CompuServe Inc. (a subsidiary of H&R Block) announced several new services available by calling the CompuServe Information Service from your computer: Travel America Club (travel service), Fifth Avenue Shopper (specialties), Showtime Entertainment Cable Information Service, and Bacchus Data Service (wine selection).

Fifth Avenue Shopper lets you arrange with FTD or Teleflora to send flowers anywhere in the world, or chocolates from Godiva Chocolatier. The Book Loop offers books and services, including assistance in tracking down rare books. Bacchus Data can recommend the perfect wine for any occasion. Showtime Entertainment Cable provides cable TV program listings and reviews.

Travel America Club provides 24-hour access to full travel services from Denver-based Travel America to make plane and hotel reservations and rental car arrangements. Tickets are mailed to your home or office, or to the airport. Reservations are confirmed within six hours. The service can also maintain a profile of your travel preferences, such as rental car agencies, hotels, airlines, special diet requirements, etc.

In addition, CompuServe announces The Communications Industry Forum, a conference for media professionals with topics ranging from the technical side of broadcast radio and television to the aesthetics of advertising, marketing and promotion. Also, CompuServe has organized a special interest group (SIG) for "electronic cottagers," or people who work at home.

Circle number 71 on the reader service card.

SITENET: Geo-Economic Data

Conway Data, Inc., announced SITENET, a free on-line data base designed to deliver site data instantly to industrial facility planners and real estate executives. SITENET's files include extensive information on tax incentives for industrial development, over 5,000 key contacts in area economic development, a directory of office and industrial parks, and data on thousands of industrial expansion projects reported each year.

Circle number 72 on the reader service card.

Accessories

NEC Printer Ribbons

NEC Home Electronics announced the 8023A ribbon cartridge for many dot matrix printers including the NEC PC-8023A, a bi-directional 80-column dot matrix printer that runs at 120 cps (characters per second), and popular models from Apple and DEC.

NEC claims the ribbon has shown a 38% longer lifespan in tests, due to a patented micro-dispersion matrix formula ink ground with porcelain stone (instead of stainless steel) to insure a fine grind without contamination. For this and many other design reasons, including a lock-out clip on the rollers to prevent direct ribbon contact until use, the ribbon prints images that do not fade or smear on conventional paper. Price: $20.

Circle number 73 on the reader service card.

The IBM PC Faire

August 26-28, 1983 Civic Auditorium & Brooks Hall San Francisco

"Jim Warren's IBM PC Faire will make a major positive contribution to the IBM Personal Computer industry and to the microcomputer user community, as have his annual West Coast Computer Faires of the past half decade. I recommend attending, exhibiting and speaking at the IBM PC Faire in San Francisco, next August.'

—Portia Isaacson, President
Future Computing, Inc.

Computer Faire 345 Swett Road Woodside CA 94062 (415) 851-7077

User's Guide

National Advertising Sales Representatives:

Northeast (617) 444-3946 Silvio Mandino
Hajar Associates
280 Hillside Ave.
Needham Heights, MA 02194

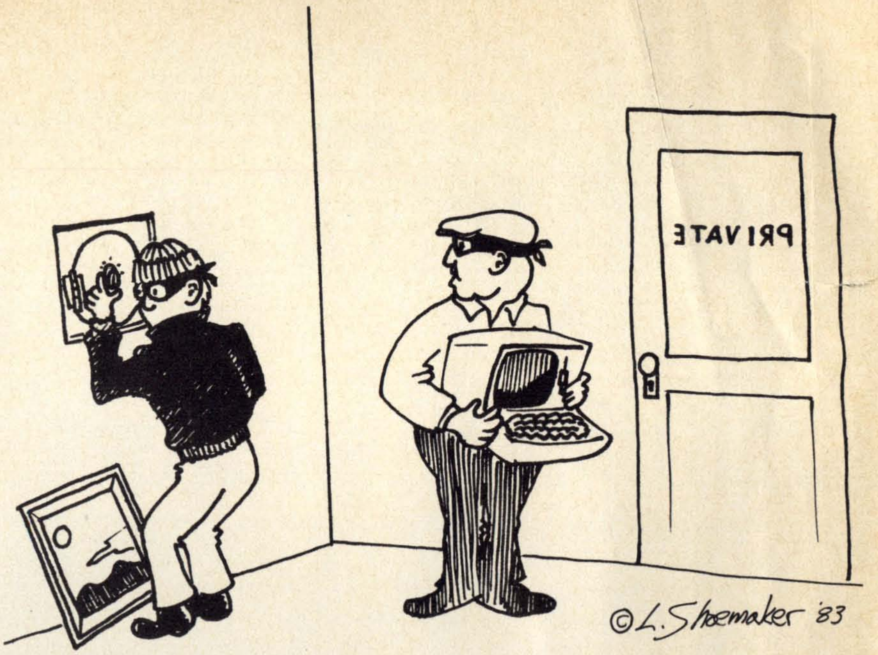
Mid Atlantic (201) 741-7744 Pamela S. Griffith
Hajar Associates
321 Broad St.
Red Bank, NJ 07701

Southeast (305) 628-3525 Beverly Ellis
Hajar Associates
Diplomat Bldg.
5400 Diplomat Circle, Suite 272
Orlando, FL 32810

Midwest (312) 966-0160 Beth Burns-Zoeller
Hajar Associates
5225 Old Orchard Rd., Suite 50
Skokie, IL 60076

Northwest (415) 964-0706 Judith Blake
Hajar Associates
1000 Elwell Ct., Suite 124
Palo Alto, CA 94303

Southwest (714) 540-3554 Linda Alcott
Hajar Associates
3303 Harbor Blvd., Suite H-4A
Costa Mesa, CA 92626



"C'mon! We got what we came for!"

User's Guide (ISSN applied for) is published six times a year by TUG, Inc., Box 3050, Stanford, CA 94305. Application to mail at second-class postage rates is pending at Woodside-Redwood City CA.

Entire contents Copyright © 1983 by TUG, Inc. All rights reserved. Reproduction of material appearing in *User's Guide* is forbidden without explicit permission. Send all requests to The Editors, Box 3050, Stanford, CA 94305.

SUBSCRIPTION INFORMATION:
Single' copy price $4.00 ($5.50 sent first-class). Subscription for one year is: U.S. — $18 if payment accompanies order, $21 if subscription is billed. Add $10 for first-class delivery.

FOREIGN - $24 U.S. for one year (surface mail) if payment accompanies order, $29 U.S. if subscription is billed. Inquire about rates for airmail delivery outside the U.S.

Inquiries about bulk, retail sale, and group rates are encouraged. Address subscription inquiries to *User's Guide*, Box 3050, Stanford, CA 94305. Allow 6-8 weeks for processing of order.

POSTMASTER: Please send address changes (Form 3579) to User's Guide, Box 3050, Stanford CA 94305.

ADVERTISE IN *USER'S GUIDE*

- **Reach an exclusive audience. *User's Guide* readers own/use CP/M systems, software, peripherals and accessories.**
- **Excellent residual pulling power. Stores carry back issues of *User's Guide* because they are useful for the life of the products described. *User's Guide* issues are used again and again by both new and experienced users.**
- **Low Rates combined with fast growing circulation. *User's Guide* #1, 10,275; #2, 20,000; #3, 25,000 ...**

For a media kit with current advertising rates, please contact: Judy Blake, **Hajar Associate** (415) 964-0706, or **User's Guide**, P.O. Box 3050, Stanford, CA 94305.

IS YOUR NEW CP/M™ MICRO STILL INTIMIDATING YOU?

By now, you may have discovered that CP/M is a wonderful operating system. But not an easy one to work with.

Here at last is **POWER!**, a super-power-packed, user-friendly program that takes the frustration and intimidation out of using CP/M.

POWER! offers 55 prompted, easy-to-use CP/M utilities in one 15k package. It automatically numbers disk files. From a screen menu, you pick the number to Copy, Erase, Reclaim, Rename, Type, etc. Your computer feeds the file names automatically. No more typing errors. And you can even run programs from the numbered menu.

No more BDOS errors. And you don't need a system disk in any disk drive!

POWER! lets you Test and Fix bad disks. Reclaim accidentally erased files or programs. Single step through memory up or down. Search, View, Change memory or disk in a snap. See Status and File Size instantly. Verify Checksums for programs. Load or Save programs at any address. Convert Hex numbers to Decimal or Binary instantly.

Now, **POWER!** also includes a special program that lets you lock sensitive files, so that only you can access them. Without the secret PASSWORD which you can create and change at will, no prying eyes will ever know your secret file even exists.

POWER! will operate in any standard CP/M or MP/M system, including CP/M-86, IBM PC, Apple (Z80 card), Osborne, Kaypro, TRS-80 conversions, S100's including NorthStar, Vector, Morrow, CompuPro, HP, TeleVideo, etc. Available in all disk formats.

InfoWorld
Software Report Card

Power
Version 2.55

	Poor	*Fair*	*Good*	*Excellent*
Performance	☐	☐	☐	☒
Documentation	☐	☐	☒	☐
Ease of Use	☐	☐	☐	☒
Error Handling	☐	☐	☐	☒

© 1982 by Popular Computing, Inc. A subsidiary of CW Communications Inc., Framingham MA. Reprinted from InfoWorld.

It comes with a 120-page easy-read guide. And at only $3.00 per utility command, **POWER!** is your best buy in software. It's 55 friendly programs rolled into one to simply put you in control of CP/M.

Don't be intimidated by CP/M. Call or send in your order today and **Take The POWER! Trip.**

Only $169. Money Back Guarantee

Charge & COD Orders Welcome VISA MasterCard

TOLL FREE (800)428-7825 Ext. 96F
IN CA: (800)428-7824 Ext. 96F
DEALERS AND OEM'S (415)567-1634 Ext.F

Take The 'POWER!' Trip.

COMPUTING!
2519 Greenwich San Francisco, CA 94123
TOLL FREE (800)428-7825 Ext. 96F
IN CA: (800)428-7824 Ext. 96F
DEALERS AND OEM'S (415)567-1634 Ext.F

ONLY $169. Calif. add 6½% sales tax
☐ **CP/M $169.** ☐ **CP/M-86 $169.** ☐ **MP/M $249.**

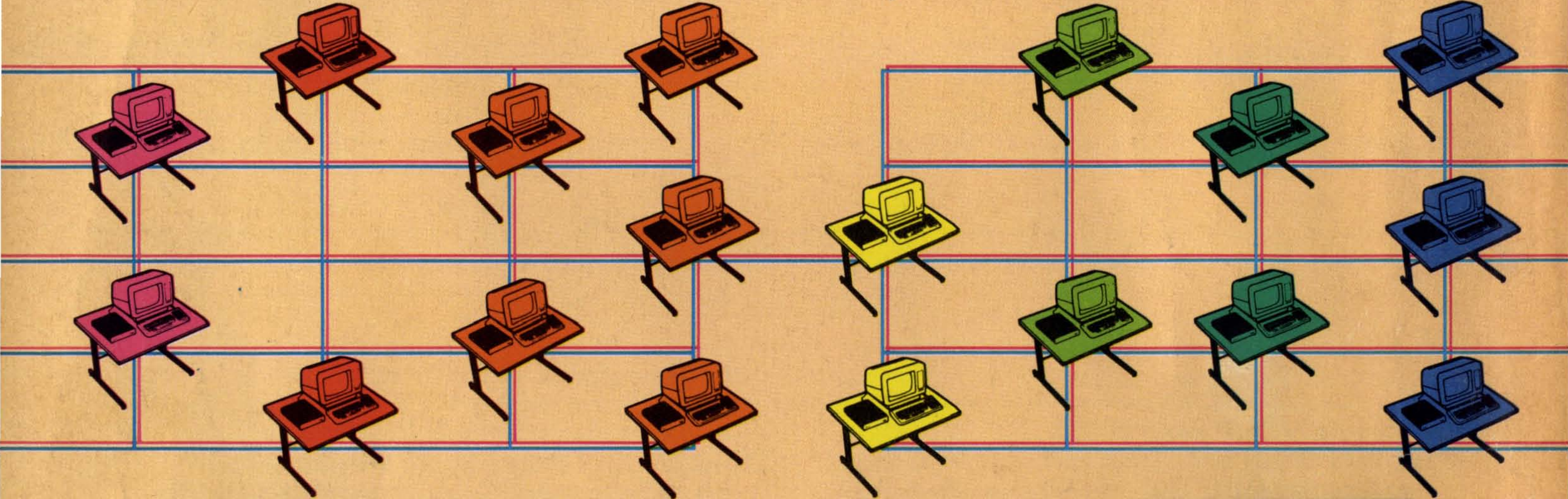
Card No. ____________
Exp. Date ____________
Computer ____________
Your Name ____________
Company Name ____________
Address ____________
City/State/Zip ____________

Circle reader service number 5

Alspa

ZERO

ANNOUNCING THE HIGH SPEED, LOW COST NETWORK



The **ZERO** is designed to bring high performance LOCAL AREA NETWORKING to users at budget prices.

The **ZERO** and **ZERO-NET** are unique. Any **ZERO** station can be a Network Master or Network Remote, permitting, for the first time, a low cost **non stop** network.

The **ZERO-NET** features a High Level Data Link Controller (message synchronous) at 400K bps carried over a simple twisted pair cable. To achieve maximum speed and reliability we use collision detection/avoidance circuitry and automatic CRC error detection/retransmission.

Each **ZERO** computer in the net can have Floppy Disk and/or Winchester Drives. The **ZERO** itself is a Z80 based Microcomputer with 64K Ram, 2K to 16K of EPROM, 2 Serial ports, 2 parallel ports, floppy disk controller, Z80CTC counter-timer and Z80DMA direct memory access.

The **ZERO** hardware design was optimized for TURBODOS*, (CP/M**, MP/M** compatible) including such enhancements as console type-ahead (buffering), 1.416 Mbytes per 8 inch double-sided floppy, multi-processing (background processing) such as print spooling, etc.

KEY PARAMETERS

- Local Area Networks up to 256 nodes per NET, with any mix of Master and Remote stations. Each station may support up to 16 logical drives, local or remote.
- Local Area Networks may be linked through gateways.
- Per Node — 0 to 2 floppies and 0 to 4 hard disks with appropriate Driver Modules.
- Per Node — parallel and/or serial printer.
- Each user can control print routing and/or spooling.
- Each node may reference a file system and/or printer on any other node.
- Each node may have an Autostart Log-on with security access protection.
- Each node may have a FIFO type Electronic Mailbox.
- Each node may operate with MP/M compatible file/record interlocks, or with special TURBODOS relaxation rules.
- Maximum recommended buss length of 4,000 L.Ft.
- Full CP/M and MP/M compatibility.
- The TURBODOS operating system can support up to one GIGA Byte (1,000 Mega Bytes) per logical drive.

The **ZERO-NET** product family — the **ZERO**, the **ZERO/FD**, the **Z-DRIVE** and the **ZNT** terminal — all the components required to configure an entire system are available for immediate delivery. The **ZERO** provides a microcomputer network that OUTPERFORMS many multi-user minicomputer systems at substantially lower cost. While the competition is still studying it, **ALSPA** has done it!

ZERO-NET

Alspa Computer, Inc.

300 Harvey West Boulevard, Santa Cruz, CA 95060
(408) 429-6000 Telex 176279

Circle reader service number 2

---

*TURBODOS is a trademark of Software 2000, Inc.

**CP/M and MP/M are trademarks of Digital Research, Inc.